



Connect:Express® Windows

Guide de programmation

Version 3.0.6.002

Sterling Commerce
An IBM Company

Connect:Express Windows
Guide de programmation

Version 3.0.6.002
Première édition

La présente documentation a pour objet d'aider les utilisateurs autorisés du système Connect:Express (ci-après le « Logiciel de Sterling Commerce »). Le Logiciel de Sterling Commerce, la documentation correspondante ainsi que les informations et le savoir-faire qu'il contient, sont la propriété de Sterling Commerce Inc. et sont confidentiels. Ils constituent des secrets commerciaux de cette dernière, de ses sociétés affiliées ou de ses/leurs concédants (ci-après dénommés collectivement « Sterling Commerce »). Ils ne peuvent pas être utilisés à des fins non autorisées ni divulgués à des tiers sans l'accord écrit préalable de Sterling Commerce. Le Logiciel de Sterling Commerce ainsi que les informations et le savoir-faire qu'il contient ont été fournis conformément à un contrat de licence qui inclut des interdictions et/ou des limitations quant à la copie, la modification et l'utilisation. La reproduction, en tout ou partie, si et lorsqu'elle est autorisée, devra inclure la présente notice d'information et la légende de copyright de Sterling Commerce Inc. Lorsqu'un Logiciel de Sterling Commerce ou un Logiciel Tiers est utilisé, reproduit ou divulgué par ou à une administration des Etats-Unis ou un cocontractant ou sous-traitant d'une telle administration, le Logiciel est assorti de DROITS LIMITES tels que définis au Titre 48 CFR 52.227-19 et est régi par les dispositions suivantes : Titre 48 CFR 2.101, 12.212, 52.227-19, 227-7201 à 227.7202-4, FAR 52.227-14 (g) (2) (6/87) et FAR 52.227-19 (c) (2) et (6/87), et le cas échéant, la licence habituelle de Sterling Commerce, tel que cela est décrit au Titre 48 CFR 227-7202-3 concernant les logiciels commerciaux et la documentation des logiciels commerciaux, y compris le DFAR 252-227-7013 (c) (1), 252.227-7015 (b) et (2), DFAR 252.227-7015 (b) (6/95), DFAR 227.7202-3 (a), selon le cas.

Le Logiciel de Sterling Commerce et la documentation correspondante sont concédés « EN L'ETAT » ou assortis d'une garantie limitée, telle que décrite dans le contrat de licence de Sterling Commerce. A l'exception des garanties limitées accordées, AUCUNE AUTRE GARANTIE EXPRESSE OU IMPLICITE N'EST CONCEDEE, Y COMPRIS LES GARANTIES DE QUALITE MARCHANDE ET DE CONVENANCE A UN USAGE PARTICULIER. La société Sterling Commerce concernée se réserve le droit de revoir cette publication périodiquement et d'effectuer des modifications quant à son contenu, sans obligation d'en informer qui que ce soit, personne physique ou personne morale. Les références faites dans le présent manuel aux produits, logiciels ou services Sterling Commerce ne signifient pas que Sterling Commerce a l'intention de les commercialiser dans tous les pays dans lesquels elle a des activités.

Imprimé aux Etats-Unis.

Copyright © 2004,2010. Sterling Commerce, Inc. Tous droits réservés.

Connect:Express est une marque déposée de Sterling Commerce. Les noms des Logiciels Tiers sont des marques ou des marques déposées de leurs sociétés respectives. Tous (toutes) autres marques ou noms de produit sont des marques ou des marques déposées de leurs sociétés respectives.

TABLE DES MATIERES

I INTRODUCTION	6
1.1 GENERALITES.....	6
1.2 CLE D'AUTORISATION LOGICIELLE.....	7
1.3 FICHER D'INITIALISATION.....	7
1.4 INTERFACES RESEAU.....	8
1.5 INSTALLATION.....	8
II L'API DE CONNECT:EXPRESS	9
2.1- ARCHITECTURE.....	9
2.2 INTEGRATION DANS UNE APPLICATION.....	10
2.2.1 Dans une application C/C++.....	10
2.2.1 Dans une application Visual Basic.....	10
III DESCRIPTION DES FONCTIONS	11
3.1 LES FONCTIONS DE SERVICE.....	12
3.1.1 Initialisation de l'API (ApiCxInit).....	12
3.1.2 Terminaison de l'API (ApiCxTerm).....	13
3.2 LES FONCTIONS CLIENT/SERVEUR.....	14
3.2.1 Accès à un moniteur Connect:Express.....	14
3.2.1.2 Connexion au moniteur (ApiCxNetConn).....	14
3.2.1.2 Déconnexion du moniteur (ApiCxNetDisc).....	15
3.2.2 Accès au répertoire des partenaires.....	16
3.2.2.1 Liste des partenaires (ApiCxNetListPart).....	16
3.2.2.2 Visualisation d'un partenaire (ApiCxNetViewPart).....	17
3.2.2.3 Visualisation d'un partenaire (Extension) (ApiCxNetViewPartEx).....	19
3.2.2.4 Ajout/Modification d'un partenaire (ApiCxNetUpdatePart).....	21
3.2.2.5 Ajout/Modification d'un partenaire (Extension) (ApiCxNetUpdatePartEx).....	23
3.2.2.4 Suppression d'un partenaire (ApiCxNetDelPart).....	25
3.2.3 Accès au répertoire des fichiers.....	26
3.2.3.1 Liste des fichiers (ApiCxNetListFile).....	26
3.2.3.2 Visualisation d'un fichier (ApiCxNetViewFile).....	27
3.2.3.3 Visualisation d'un fichier (Extension) (ApiCxNetViewFileEx).....	29
3.2.3.4 Ajout/Modification d'un fichier (ApiCxNetUpdateFile).....	31
3.2.3.5 Ajout/Modification d'un fichier (Extension) (ApiCxNetUpdateFileEx).....	32
3.2.3.6 Suppression d'un fichier (ApiCxNetDelFile).....	34
3.2.4 Accès au répertoire des clients.....	35
3.2.4.1 Liste des clients (ApiCxNetListClient).....	35
3.2.4.2 Visualisation d'un client (ApiCxNetViewClient).....	36
3.2.4.3 Ajout/Modification d'un client (ApiCxNetUpdateClient).....	37
3.2.4.4 Suppression d'un client (ApiCxNetDelClient).....	38
3.2.5 Accès au répertoire des paramètres des clients SSL.....	39
3.2.5.1 Liste des clients SSL (ApiCxNetListSslparmCli).....	39
3.2.5.2 Visualisation d'un client SSL (ApiCxNetViewSslparmCli).....	40
3.2.5.3 Visualisation d'un client SSL (ApiCxNetViewSslparmCliEx).....	41
3.2.5.4 Ajout/Modification d'un client SSL (ApiCxNetUpdateSslparmCli).....	41
3.2.5.5 Ajout/Modification d'un client SSL (ApiCxNetUpdateSslparmCliEx).....	42
3.2.5.6 Suppression d'un client SSL (ApiCxNetDelSslparmCli).....	43
3.2.6 Accès à l'activité du moniteur de transferts.....	44
3.2.6.1 Liste de noms autorisés (ApiCxNetListAuth).....	44
3.2.6.2 Détail d'un fichier autorisé (ApiCxNetViewAuth).....	45
3.2.6.3 Type de définition pour un partenaire (ApiCxNetPartAuth).....	46
3.2.6.4 Soumission d'une requête de transfert (ApiCxNetSubmitReq).....	47
3.2.6.5 Soumission d'une requête de transfert (Extension) (ApiCxNetSubmitReqEx).....	48
3.2.6.6 Liste des requêtes de transfert (ApiCxNetListReq).....	50
3.2.6.7 Visualisation d'une requête de transfert (ApiCxNetViewReq).....	52
3.2.6.8 Action sur une requête de transfert (ApiCxNetActionReq).....	53

3.2.7 Accès aux messages du moniteur de transferts.....	54
3.2.7.1 Liste des messages (ApiCxNetListLog).....	54
3.2.8 Accès au journal des transferts du moniteur de transferts.....	55
3.2.7.1 Liste du journal (ApiCxNetListJnl).....	55
3.2.8.2 Visualisation d'un enregistrement du journal (ApiCxNetViewJnl).....	57
3.2.9 Accès aux notifications du moniteur de transferts.....	60
3.2.9.1 Liste des notifications (ApiCxNetListNot).....	60
3.2.9.2 Visualisation d'une notification (ApiCxNetViewNot).....	62
3.2.9.3 Action sur une notification (ApiCxNetActionNot).....	63
3.2.10 Accès aux tables du moniteur.....	64
3.2.10.1 Liste des tables de Présentation PeSIT (ApiCxNetListPPres).....	64
3.2.10.2 Visualisation d'une table de Présentation PeSIT (ApiCxNetViewPPres).....	65
3.2.10.3 Ajout/Modification d'une table de Présentation PeSIT (ApiCxNetUpdatePPres).....	66
3.2.10.4 Suppression d'une table de Présentation PeSIT (ApiCxNetDelPPres).....	67
3.2.10.5 Liste des tables de Session PeSIT (ApiCxNetListPSess).....	68
3.2.10.6 Visualisation d'une table de Session PeSIT (ApiCxNetViewPSess).....	69
3.2.10.7 Ajout/Modification d'une table de Session PeSIT (ApiCxNetUpdatePPres).....	70
3.2.10.8 Suppression d'une table de Session PeSIT (ApiCxNetDelPSess).....	71
3.2.10.9 Liste des tables de Présentation ETEBAC-3 (ApiCxNetListPEtb3).....	72
3.2.10.10 Visualisation d'une table de Présentation ETEBAC-3 (ApiCxNetViewPEtb3).....	73
3.2.10.11 Ajout/Modification d'une table de Présentation ETEBAC-3 (ApiCxNetUpdatePEtb3).....	74
3.2.10.12 Suppression d'une table de Présentation ETEBAC-3 (ApiCxNetDelPEtb3).....	75
3.2.11 Accès aux paramètres du moniteur.....	76
3.2.11.1 Visualisation des paramètres de Démarrage du moniteur (ApiCxNetViewPrmStr).....	76
3.2.11.2 Visualisation des paramètres de Service du moniteur (ApiCxNetViewPrmSrv).....	77
3.2.11.3 Visualisation des paramètres Fichier du moniteur (ApiCxNetViewPrmFil).....	78
3.2.11.4 Visualisation des paramètres d'Autorisation du moniteur (ApiCxNetViewPrmAut).....	79
3.2.11.5 Visualisation des éléments de la clé logicielle (Licence) (ApiCxNetListLicense).....	79
3.2.11.6 Visualisation des paramètres de notification du moniteur (ApiCxNetViewPrmNot).....	80
3.2.11.7 Visualisation des paramètres TCP/IP du moniteur (ApiCxNetViewPrmIp).....	81
3.2.11.8 Visualisation des paramètres SNA LU6.2 du moniteur (ApiCxNetViewPrmSna).....	82
3.2.11.9 Visualisation des paramètres X.25 du moniteur (ApiCxNetViewPrmX25).....	83
3.2.11.10 Visualisation des paramètres Canal Nommé du moniteur (ApiCxNetViewPrmNp).....	84
3.2.11.11 Liste des serveurs SSL (ApiCxNetListSslparmSrv).....	85
3.2.11.12 Visualisation des paramètres d'un serveur SSL (ApiCxNetViewSslparmSrv).....	86
3.2.11.13 Visualisation des paramètres d'un serveur SSL (ApiCxNetViewSslparmSrvEx).....	86
3.3 LES FONCTIONS D'ACCES DIRECT.....	88
3.3.1 Paramètres de Démarrage du moniteur.....	88
3.3.1.1 Visualisation des paramètres de Démarrage du moniteur (ApiCxShrViewPrmStr).....	88
3.3.1.2 Modification des paramètres de Démarrage du moniteur (ApiCxShrUpdatePrmStr).....	89
3.3.2 Paramètres de Service du moniteur.....	90
3.3.2.1 Visualisation des paramètres de Service du moniteur (ApiCxShrViewPrmSrv).....	90
3.3.2.2 Modification des paramètres de Service du moniteur.....	91
3.3.3 Paramètres Fichier du moniteur.....	92
3.3.3.1 Visualisation des paramètres Fichier du moniteur (ApiCxShrViewPrmFil).....	92
3.3.3.2 Modification des paramètres Fichier du moniteur (ApiCxShrUpdatePrmFil).....	93
3.3.4 Paramètres d'Autorisation du moniteur.....	94
3.3.4.1 Visualisation des paramètres d'Autorisation du moniteur (ApiCxShrViewPrmAut).....	94
3.3.4.2 Modification des paramètres d'Autorisation du moniteur (ApiCxShrUpdatePrmAut).....	95
3.3.4.3 Chargement du fichier license.key par le moniteur (ApiCxNetLoadLicense).....	95
3.3.5 Paramètres de Notification du moniteur.....	97
3.3.5.1 Visualisation des paramètres de Notification du moniteur (ApiCxShrViewPrmNot).....	97
3.3.5.2 Modification des paramètres de Notification du moniteur (ApiCxShrUpdatePrmNot).....	98
3.3.6 Paramètres TCP/IP du moniteur.....	99
3.3.6.1 Visualisation des paramètres TCP/IP du moniteur (ApiCxShrViewPrmIp).....	99
3.3.6.2 Modification des paramètres TCP/IP du moniteur (ApiCxShrUpdatePrmIp).....	100
3.3.7 Paramètres SNA/LU6.2 du moniteur.....	101
3.3.7.1 Visualisation des paramètres SNA/LU6.2 du moniteur (ApiCxShrViewPrmSna).....	101
3.3.7.2 Modification des paramètres SNA/LU6.2 du moniteur (ApiCxShrUpdatePrmSna).....	102
3.3.8 Paramètres X.25 du moniteur.....	103
3.3.8.1 Visualisation des paramètres X.25 du moniteur (ApiCxShrViewPrmX25).....	103
3.3.8.2 Modification des paramètres X.25 du moniteur (ApiCxShrUpdatePrmX25).....	104

3.3.9 Paramètres Canal Nommé du moniteur	105
3.3.9.1 Visualisation des paramètres Canal Nommé du moniteur (ApiCxShrViewPrmNp)	105
3.3.9.2 Modification des paramètres Canal Nommé du moniteur (ApiCxShrUpdatePrmNp).....	106
3.3.10 Paramètres des serveurs SSL.....	107
3.3.10.1 Liste des serveurs SSL (ApiCxShrListSslparmSrv)	107
3.3.10.2 Visualisation des paramètres d'un serveur SSL (ApiCxShrViewSslparmSrv).....	108
3.3.10.3 Visualisation des paramètres d'un serveur SSL (ApiCxShrViewSslparmSrvEx)	108
3.3.10.4 Modification des paramètres d'un serveur SSL (ApiCxShrUpdateSslparmSrv)	109
3.3.10.5 Modification des paramètres d'un serveur SSL (ApiCxShrUpdateSslparmSrvEx).....	110
3.3.10.6 Suppression d'un serveur SSL (ApiCxShrDelSslparmSrv)	110

I Introduction

1.1 Généralités

Ce document présente l'interface de programmation (API) de **Connect:Express Windows**. Il décrit les techniques permettant à une application de paramétrer, mettre en œuvre les transferts et d'en effectuer leur suivi, avec un ou plusieurs moniteurs de transferts Connect:Express.

Les fonctions de l'API sont présentées en deux groupes fonctionnels :

- *Les fonctions d'administration*, accessibles en local avec le produit standard ou, à distance, avec l'option Activitiy Manager,
- *Les fonctions de mise en œuvre et de suivi des transferts*, accessibles en local ou à distance avec le produit standard.

Les fonctions de l'API effectuent soit directement la mise à jour du fichier d'initialisation du moniteur, pour le paramétrage, soit établissent une connexion Client/Serveur sur un réseau TCP/IP ou Canal Nommé, pour l'ensemble des autres fonctions.

L'utilisation de l'API est soumis à des contrôles d'autorisation, au niveau de l'API elle même et au niveau du moniteur Connect:Express.

1.2 Clé d'autorisation logicielle

Le produit Connect:Express est livré avec un fichier license.key contenant une clé d'autorisation qui permet au moniteur de transferts et à l'API d'effectuer les contrôles nécessaires à leur utilisation, en fonction de la licence acquise.

Les contrôles effectués par l'API concernent :

- sa localisation : en standard les fonctions d'administration sont accessibles localement, les fonctions de mise en œuvre et de suivi sont accessibles localement et à distance
- le nombre de moniteurs définis et le nombre de moniteurs connectés simultanément : en standard un seul moniteur peut être défini et une seule connexion peut exister à un instant donné par une même instance de l'API.

Les contrôles effectués par le moniteur concernent :

- la localisation du client: en fonction du nombre de clients distants de la clé sont acceptés seuls les clients locaux, ou, les clients locaux et les clients distants
- le type d'API qui se connecte: Standard, Client/Serveur ou Activity manager
- Le nombre de clients distants connectés en parallèle

Le fichier d'autorisation license.key doit être placé dans le répertoire d'exécution de l'API.

En fonction de cette clé, un ou plusieurs moniteurs peuvent être connectés simultanément et l'accès aux fonctions d'administration est permis ou non, localement ou à distance.

1.3 Fichier d'initialisation

Par ailleurs, l'API utilise un fichier d'initialisation contenant des paramètres de démarrage. Par défaut, l'API utilise le fichier 'TOMNT.INI' situé dans son répertoire d'exécution et il peut être modifié par l'application.

Pour les options Client/Serveur et Activity manager, le fichier d'initialisation est le fichier 'IUTOM.INI' de l'option.

La description des informations du fichier d'initialisation de l'API est donnée dans le '**Guide de l'Installation et des Utilitaires de Connect:Express Windows**'

1.4 Interfaces réseau

L'API 32 bits s'interface sur les réseaux:

- TCP/IP
- Canal nommé (Named Pipe)

Le réseau utilisé doit être activé au niveau du moniteur auquel on doit se connecter, et les informations réseau s'y rapportant (Numéro de port IP, nom du canal nommé...) doivent être identiques à celles fournies à l'API lors de l'établissement de la connexion Client/Serveur.

Le moniteur auquel on se connecte au travers de l'API gère un délai d'absence de trafic (time out), paramétrable au niveau du moniteur. Si le délai est atteint, la connexion est rompue par le moniteur.

1.5 Installation

L'API est localisée sur le CD-ROM dans le répertoire SDKwin32 et contient les fichiers suivants :

- Apicxv3.dll : la librairie dynamique
- Apicxv3.lib : la librairie d'import
- Apicxv3d.h : les déclarations des structures et des constantes en langage C
- Apicxv3p.h : les prototypes des fonctions en langage C
- Apicxv3.txt : déclarations et prototypes pour Visual Basic
- TOMREQ.C : exemple d'implémentation de l'API en langage C

Pour installer l'API, recopier ces fichiers dans un répertoire de travail, manuellement. Voir l'utilisation du fichier d'initialisation indiqués ci-avant.

II L'API de Connect:Express

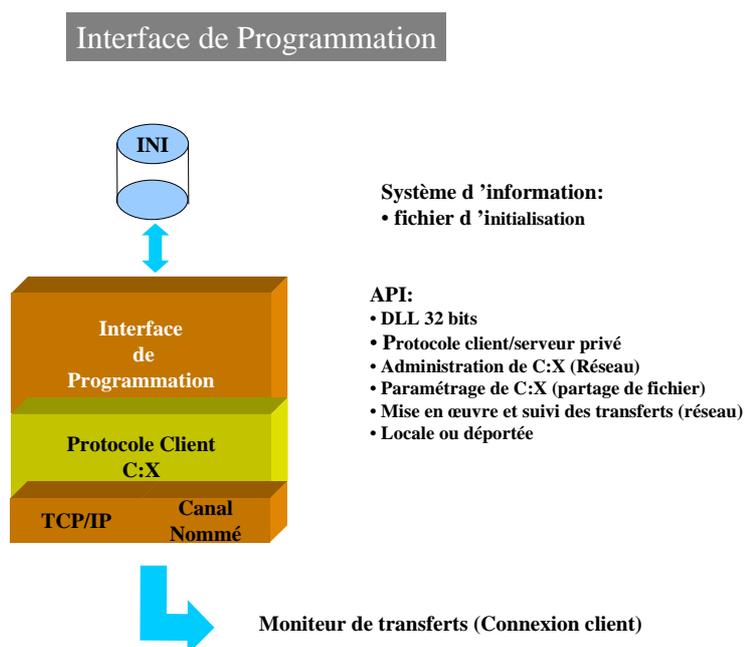
2.1- Architecture

L'API permet à une application Windows 32 bits de communiquer avec le moniteur de transferts Connect:Express et d'accéder à son système d'informations au travers d'une connexion réseau ou par le système de partage de fichiers.

La connexion, via le protocole TCP/IP ou un Canal Nommé (Named Pipe) et l'établissement d'un dialogue Client/Serveur permet :

- l'accès en mise à jour aux répertoires et aux tables
- l'initialisation de transferts
- le suivi de transferts
- la consultation des messages et du journal
- l'accès aux notifications mémorisées
- la consultation des paramètres du moniteur

Le partage de fichier permet de mettre à jour les paramètres du moniteur.



L'utilisation de l'API s'effectue après identification du CLIENT utilisateur et sous le contrôle de ses droits d'accès définis dans le répertoire des clients du moniteur Connect:Express.

2.2 Intégration dans une application

2.2.1 Dans une application C/C++

L'intégration de l'API dans une application C/C++ peut s'effectuer de deux manières :

- **STATIQUE** : l'application doit intégrer, lors de l'édition des liens, la librairie d'import 'APICXV3.LIB'.
- **DYNAMIQUE** : l'application doit charger dynamiquement la librairie 'APICXV3.DLL' et récupérer l'adresse des fonctions pour les mettre en œuvre.

La convention d'appel des fonctions de l'API est de type **stdcall**

Le fichier 'APICXV3D.H' contient les définitions, en langage C, des structures et des constantes à indiquer lors de l'appel des fonctions de cette API.

Le fichier 'APICXV3P.H' contient les prototypes, en langage C des fonctions de cette API. En C++, l'utilisation de ce fichier doit se faire de la manière suivante :

```
extern "C"  
{  
    #include "apicxv3p.h"  
}
```

2.2.1 Dans une application Visual Basic

Le fichier 'APICXV3.TXT' contient les déclarations à utiliser dans une application Visual Basic.

Ce fichier est à utiliser avec la 'Visionneuse d'API' qui permet de récupérer les types de données et les déclarations des fonctions de cette API.

III Description des fonctions

Lors de l'appel aux fonctions de l'API, l'application indique une adresse de structure.

Cette structure, spécifique à chaque fonction doit être initialisée de la manière suivante et avant chaque appel:

- Remplir la structure avec des espaces sur toute sa longueur
- Renseigner les champs nécessaires en entrée avec des valeurs cohérentes (utiliser les constantes définies dans le fichier 'APICXV3D.H')

Les tableaux de caractères doivent être complétés par des espaces sauf quand il est précisé que le tableau est une chaîne de caractères (null terminated), dans ce cas il est complété par des zéros binaires

En retour, l'API complète cette structure en indiquant :

- Les codes retour d'exécution de la fonction par le moniteur
- Les données liées à cette exécution

Et rend un code retour direct.

Le code retour direct d'appel de la fonction de l'API indique que la commande a bien été exécutée ou non.

Les codes retour ApiRC, SysRc, TcpRc, et TomRc permettent de préciser, le cas échéant, le type d'erreur rencontré par l'API ou par le moniteur lors de l'exécution de la commande.

Entre deux appels aux fonctions de l'API et pendant une connexion à un moniteur, un temps maximum de silence doit être respecté sous peine de voir la communication fermée par le moniteur Connect:Express.

Ce délai est indiqué dans les paramètres de Lancement du Moniteur connecté.

Trois groupes de fonctions sont accessibles:

- Les fonctions de service (ApiCxInit et ApiCxTerm)
- Les fonctions utilisant la connexion Client/Serveur (ApiCxNet...)
- Les fonctions accédant directement au fichier d'initialisation du moniteur par le système de partage de fichiers (ApiCxShr...)

3.1 Les fonctions de Service

3.1.1 Initialisation de l'API (ApiCxInit)

Description

Cette fonction doit être la première à être appelée par l'application utilisatrice de l'API et ne doit être appelée qu'une seule fois.

Elle permet à l'API de se réserver les ressources nécessaires à son exécution et de contrôler le numéro d'autorisation présent dans le fichier license.key.

En entrée

L'application indique :

- *IniFName* : le nom complet du fichier d'initialisation dans lequel l'API doit trouver les informations suivantes :
 - ✓ L'indicateur de mise en œuvre du protocole TCP/IP
 - ✓ La valeur de l'échéance de temporisation pour les connexions Client/Serveur

En sortie

Le code retour direct de la fonction indique :

- -1 : l'adresse de la structure APICX_INIT donnée par l'application est incorrecte
- 0 : la fonction a échoué, voir le code retour API
- 1 : la fonction a été exécutée correctement

Les codes *ApiRc*, *SysRc* et *TcpRc* sont documentés dans le *Guide utilisateur de Connect:Express Windows*.

Si la fonction a été exécutée correctement, les champs suivants sont retournés :

- *AuthVer* : indique le type de produit autorisé par le numéro.
- *MaxCx* : nombre maximum de connexions simultanées à des moniteurs (1 à 8)
- *MaxMon* : nombre maximum de moniteurs définis dans l'interface graphique (1 à 255)
- *FlagIp* : O (Oui) ou N (Non) pour indiquer la mise en œuvre de TCP/IP ou non
- *MonAuth* : la liste des moniteurs Connect:Express **auxquels** l'application peut se connecter

3.1.2 Terminaison de l'API (ApiCxTerm)

Description

Cette fonction doit être la dernière à être appelée par l'application utilisatrice de l'API et rend les autres fonctions inutilisables sans passer par la fonction d'initialisation au préalable.

Elle permet à l'API de libérer les ressources allouées pour son exécution lors de son initialisation.

En entrée

Pas de paramètres.

En sortie

Le code retour direct de la fonction indique :

- -1 : l'adresse de la structure APICX_TERM donnée par l'application est incorrecte
- 0 : la fonction a échoué, voir le code retour API
- 1 : la fonction a été exécutée correctement

Les codes *ApiRc* sont documentés dans le *Guide utilisateur de Connect:Express Windows*.

3.2 Les fonctions Client/Serveur

3.2.1 Accès à un moniteur Connect:Express

3.2.1.2 Connexion au moniteur (ApiCxNetConn)

Description

Cette fonction permet à l'application d'établir le dialogue Client / Serveur avec un moniteur Connect:Express sur un lien de type TCP/IP ou Canal Nommé.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique les informations réseau nécessaires à l'ouverture de la communication avec le moniteur :

- *Linktp* : type de lien réseau (TCP/IP ou Canal Nommé)
- *IpAddr* : pour le réseau TCP/IP – Adresse IP du système ou le moniteur C :X s'exécute
- *IpName* : pour le réseau TCP/IP - Nom d'hôte TCP/IP du système ou le moniteur C :X s'exécute
- *IpPort* : pour le réseau TCP/IP – Numéro de port IP sur lequel le moniteur C :X attend les connexions Client
- *NpName* : Pour le réseau Canal Nommé – Nom du canal nommé créé par le moniteur C :X pour attendre les connexions Client
- *CliName* : nom du client qui se connecte
- *CliPsw* : mot de passe du client qui se connecte

En sortie

Le code retour direct de la fonction indique :

- -1 : l'adresse de la structure APICXN_CX donnée par l'application est incorrecte
- 0 : la fonction a échoué, voir le code retour API
- 1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_CX est complétée de la manière suivante :

- *CxId* : identifiant de connexion. Cette valeur doit être reprise par l'application pour cette connexion
- *MonId* : contient trois libellés permettant d'identifier le type et la version du moniteur Connect:Express, l'identifiant de la connexion client pour le moniteur ainsi que la date et l'heure de connexion.
- *CliAuth* : indique la liste des droits du client connecté
- *ApiType* : c'est le type d'API mis en œuvre qui permet de déterminer les fonctions accessibles.
- *MonType* : c'est le type de moniteur Connect:Express auquel l'application est connectée.

3.2.1.2 Déconnexion du moniteur (*ApiCxNetDisc*)

Description

Cette fonction permet à l'application de fermer le dialogue Client / Serveur précédemment établi avec un moniteur Connect:Express sur un lien de type TCP/IP ou Canal Nommé.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *Cxld* : l'identifiant de la connexion qu'elle veut fermer.

En sortie

Le code retour direct de la fonction indique :

- -1 : l'adresse de la structure APICXN_DISC donnée par l'application est incorrecte
- 0 : la fonction a échoué, voir le code retour API
- 1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.2 Accès au répertoire des partenaires

3.2.2.1 Liste des partenaires (*ApiCxNetListPart*)

Description

Cette fonction permet à l'application d'obtenir la liste des partenaires définis au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des partenaires
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_PART donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_PART est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms symboliques de partenaire dans la liste
- *TotDef* : nombre de noms symboliques de partenaire définis dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue est composée de noms symboliques de partenaires sur 8 caractères complétés avec des espaces si nécessaire.

3.2.2.2 Visualisation d'un partenaire (ApiCxNetViewPart)

Description

Cette fonction permet à l'application de visualiser la définition d'un partenaire défini au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PART donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PART est complétée de la manière suivante :

- *PartPsw* : mot de passe du partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocName* : nom symbolique local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocPsw* : mot de passe local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocTyp* : indique si l'identifiant local est statique (APICX_LOCS) ou dynamique (APICX_LOCD).
- *Comment* : libellé décrivant la définition de ce partenaire sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du partenaire et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *PartType* : type du partenaire et peut prendre les valeurs 'APICX_PCE' pour Connect:Express ou 'APICX_POTH' pour autre
- *Restart* : permet d'autoriser les reprises automatiques 'O' ou non 'N' avec ce partenaire
- *ProtType* : type de protocole utilisé avec ce partenaire et peut prendre les valeurs 'APICX_PPD' pour PeSIT hors SIT version D, 'APICX_PPE' pour PeSIT hors SIT version E ou 'APICX_PET3' pour ETEBAC-3.
- *MaxSess* : nombre maximum total de connexions simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessIn* : nombre maximum de connexions entrantes simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessOut* : nombre maximum de connexions sortantes simultanées avec ce partenaire sur 3 caractères pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *SessName* : nom de la table de session du protocole PeSIT utilisée pour ce partenaire sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62'
- *IpAddr* : adresse IP du partenaire au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du partenaire sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *IpPort* : numéro de port TCP/IP du partenaire sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)

- *LuName* : nom de la LU du partenaire sur 8 caractères en majuscules et complété par des espaces
- *ModName* : nom du mode du partenaire sur 8 caractères en majuscules et complété par des espaces
- *TpName* : nom du programme de transaction du partenaire sur 64 caractères et complété par des espaces
- *XLAddr* : adresse locale spécifiée lors de l'appel de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XRAddr* : adresse de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XPort* : numéro de port X.25 à utiliser pour appeler ce partenaire sur 2 caractères numériques et terminé par au moins un caractère nul (\0)
- *XUdf* : champ de données utilisateur à spécifier lors de l'appel de ce partenaire sur 8 caractères en majuscules et complété par des espaces
- *XFac* : champ de facilités à spécifier lors de l'appel de ce partenaire sur 32 caractères en majuscules et complété par des espaces

3.2.2.3 Visualisation d'un partenaire (Extension) (ApiCxNetViewPartEx)

Description

Cette fonction permet à l'application de visualiser la définition d'un partenaire défini au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PART_EX donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PART_EX est complétée de la manière suivante :

- *PartPsw* : mot de passe du partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocName* : nom symbolique local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocPsw* : mot de passe local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocTyp* : indique si l'identifiant local est statique (APICX_LOCS) ou dynamique (APICX_LOCD).
- *Comment* : libellé décrivant la définition de ce partenaire sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du partenaire et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *PartType* : type du partenaire et peut prendre les valeurs 'APICX_PCE' pour Connect:Express ou 'APICX_POTH' pour autre
- *Restart* : permet d'autoriser les reprises automatiques 'O' ou non 'N' avec ce partenaire
- *ProtType* : type de protocole utilisé avec ce partenaire et peut prendre les valeurs 'APICX_PPD' pour PeSIT hors SIT version D, 'APICX_PPE' pour PeSIT hors SIT version E ou 'APICX_PET3' pour ETEBAC-3.
- *MaxSess* : nombre maximum total de connexions simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessIn* : nombre maximum de connexions entrantes simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessOut* : nombre maximum de connexions sortantes simultanées avec ce partenaire sur 3 caractères pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *SessName* : nom de la table de session du protocole PeSIT utilisée pour ce partenaire sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62'
- *IpAddr* : adresse IP du partenaire au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du partenaire sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

- *IpPort* : numéro de port TCP/IP du partenaire sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *LuName* : nom de la LU du partenaire sur 8 caractères en majuscules et complété par des espaces
- *ModName* : nom du mode du partenaire sur 8 caractères en majuscules et complété par des espaces
- *TpName* : nom du programme de transaction du partenaire sur 64 caractères et complété par des espaces
- *XLAddr* : adresse locale spécifiée lors de l'appel de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XRAddr* : adresse de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XPort* : numéro de port X.25 à utiliser pour appeler ce partenaire sur 2 caractères numériques et terminé par au moins un caractère nul (\0)
- *XUdf* : champ de données utilisateur à spécifier lors de l'appel de ce partenaire sur 8 caractères en majuscules et complété par des espaces
- *XFac* : champ de facilités à spécifier lors de l'appel de ce partenaire sur 32 caractères en majuscules et complété par des espaces
- *SslUsed* : '1' indique que SSL est utilisé pour les transferts avec ce partenaire. Autre valeur : SSL n'est pas utilisé.
- *Sslparm* : Champ de 8+1 caractère indiquant le nom symbolique d'une définition de paramètres SSL client. Le nom est terminé par un zéro binaire.
- *RemoteClientSubjectDn* : Champ de 255+1 caractères indiquant des critères d'autorisation pour le DN objet du certificat du client distant (Transfert SSL/TCP). Les critères sont terminés par un zéro binaire.
- *RemoteClientRootDn* : Champ de 255+1 caractères indiquant des critères d'autorisation pour le DN racine du certificat du client distant (Transfert SSL/TCP). Les critères sont terminés par un zéro binaire.
- *RemoteServerSubjectDn* : Champ de 255+1 caractères indiquant des critères d'autorisation pour le DN objet du certificat du serveur distant (Transfert SSL/TCP). Les critères sont terminés par un zéro binaire.
- *RemoteServerRootDn* : Champ de 255+1 caractères indiquant des critères d'autorisation pour le DN racine du certificat du serveur distant (Transfert SSL/TCP). Les critères sont terminés par un zéro binaire.

3.2.2.4 Ajout/Modification d'un partenaire (ApiCxNetUpdatePart)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un partenaire au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. Si le partenaire n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *PartPsw* : mot de passe du partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocName* : nom symbolique local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocPsw* : mot de passe local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocTyp* : indique si l'identifiant local est statique (APICX_LOCS) ou dynamique (APICX_LOCD).
- *Comment* : libellé décrivant la définition de ce partenaire sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du partenaire et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *PartType* : type du partenaire et peut prendre les valeurs 'APICX_PCE' pour Connect:Express ou 'APICX_POTH' pour autre
- *Restart* : permet d'autoriser les reprises automatiques 'O' ou non 'N' avec ce partenaire
- *ProtType* : type de protocole utilisé avec ce partenaire et peut prendre les valeurs 'APICX_PPD' pour PeSIT hors SIT version D, 'APICX_PPE' pour PeSIT hors SIT version E ou 'APICX_PET3' pour ETEBAC-3.
- *MaxSess* : nombre maximum total de connexions simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessIn* : nombre maximum de connexions entrantes simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessOut* : nombre maximum de connexions sortantes simultanées avec ce partenaire sur 3 caractères pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *SessName* : nom de la table de session du protocole PeSIT utilisée pour ce partenaire sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62'
- *IpAddr* : adresse IP du partenaire au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du partenaire sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *IpPort* : numéro de port TCP/IP du partenaire sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *LuName* : nom de la LU du partenaire sur 8 caractères en majuscules et complété par des espaces
- *ModName* : nom du mode du partenaire sur 8 caractères en majuscules et complété par des espaces
- *TpName* : nom du programme de transaction du partenaire sur 64 caractères et complété par des espaces
- *XLAddr* : adresse locale spécifiée lors de l'appel de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XRAAddr* : adresse de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XPort* : numéro de port X.25 à utiliser pour appeler ce partenaire sur 2 caractères numériques et terminé par au moins un caractère nul (\0)

- *XUdf* : champ de données utilisateur à spécifier lors de l'appel de ce partenaire sur 8 caractères en majuscules et complété par des espaces
- *XFac* : champ de facilités à spécifier lors de l'appel de ce partenaire sur 32 caractères en majuscules et complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PART donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le ***Guide utilisateur de Connect:Express Windows***. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.2.5 Ajout/Modification d'un partenaire (Extension) (ApiCxNetUpdatePartEx)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un partenaire au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. Si le partenaire n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *PartPsw* : mot de passe du partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocName* : nom symbolique local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocPsw* : mot de passe local identifiant Connect:Express pour ce partenaire sur 8 caractères en majuscules, complété par des espaces
- *LocTyp* : indique si l'identifiant local est statique (APICX_LOCS) ou dynamique (APICX_LOCD).
- *Comment* : libellé décrivant la définition de ce partenaire sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du partenaire et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *PartType* : type du partenaire et peut prendre les valeurs 'APICX_PCE' pour Connect:Express ou 'APICX_POTH' pour autre
- *Restart* : permet d'autoriser les reprises automatiques 'O' ou non 'N' avec ce partenaire
- *ProtType* : type de protocole utilisé avec ce partenaire et peut prendre les valeurs 'APICX_PPD' pour PeSIT hors SIT version D, 'APICX_PPE' pour PeSIT hors SIT version E ou 'APICX_PET3' pour ETEBAC-3.
- *MaxSess* : nombre maximum total de connexions simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessIn* : nombre maximum de connexions entrantes simultanées avec ce partenaire sur 3 caractères et pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *MaxSessOut* : nombre maximum de connexions sortantes simultanées avec ce partenaire sur 3 caractères pouvant prendre une valeur minimum de '000' et une valeur maximum de '128' ou espace pour 255
- *SessName* : nom de la table de session du protocole PeSIT utilisée pour ce partenaire sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62'
- *IpAddr* : adresse IP du partenaire au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du partenaire sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *IpPort* : numéro de port TCP/IP du partenaire sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *LuName* : nom de la LU du partenaire sur 8 caractères en majuscules et complété par des espaces
- *ModName* : nom du mode du partenaire sur 8 caractères en majuscules et complété par des espaces
- *TpName* : nom du programme de transaction du partenaire sur 64 caractères et complété par des espaces
- *XLAddr* : adresse locale spécifiée lors de l'appel de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XRAAddr* : adresse de ce partenaire sur au plus 15 caractères numériques et terminée par au moins un caractère nul (\0)
- *XPort* : numéro de port X.25 à utiliser pour appeler ce partenaire sur 2 caractères numériques et terminé par au moins un caractère nul (\0)

- *XUdf* : champ de données utilisateur à spécifier lors de l'appel de ce partenaire sur 8 caractères en majuscules et complété par des espaces
- *XFac* : champ de facilités à spécifier lors de l'appel de ce partenaire sur 32 caractères en majuscules et complété par des espaces
- *SslUsed* : '1' indique que SSL est utilisé pour les transferts avec ce partenaire. Autre valeur : SSL n'est pas utilisé.
- *Sslparm* : Champ de 8+1 caractère indiquant le nom symbolique d'une définition de paramètres SSL client. Le nom doit être terminé par un zéro binaire.
- *RemoteClientSubjectDn* : Champ facultatif de 255+1 caractères indiquant des critères d'autorisation pour le DN objet du certificat du client distant (Transfert SSL/TCP). Si présents, les critères doivent être terminés par un zéro binaire (sinon placer espace ou zéro binaire dans le premier byte).
- *RemoteClientRootDn* : Champ facultatif de 255+1 caractères indiquant des critères d'autorisation pour le DN racine du certificat du client distant (Transfert SSL/TCP). Si présents, les critères doivent être terminés par un zéro binaire (sinon placer espace ou zéro binaire dans le premier byte).
- *RemoteServerSubjectDn* : Champ facultatif de 255+1 caractères indiquant des critères d'autorisation pour le DN objet du certificat du serveur distant (Transfert SSL/TCP). Si présents, les critères doivent être terminés par un zéro binaire (sinon placer espace ou zéro binaire dans le premier byte).
- *RemoteServerRootDn* : Champ facultatif de 255+1 caractères indiquant des critères d'autorisation pour le DN racine du certificat du serveur distant (Transfert SSL/TCP). Si présents, les critères doivent être terminés par un zéro binaire (sinon placer espace ou zéro binaire dans le premier byte).

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PART_EX donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le ***Guide utilisateur de Connect:Express Windows***. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.2.4 Suppression d'un partenaire (ApiCxNetDelPart)

Description

Cette fonction permet à l'application de supprimer la définition d'un partenaire au répertoire des partenaires du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_PART donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le **Guide utilisateur de Connect:Express Windows**. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.3 Accès au répertoire des fichiers

3.2.3.1 Liste des fichiers (*ApiCxNetListFile*)

Description

Cette fonction permet à l'application d'obtenir la liste des fichiers définis au répertoire des fichiers du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des fichiers
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXN_LIST_FILE* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXN_LIST_FILE* est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms symboliques de fichier dans la liste
- *TotDef* : nombre de noms symboliques de fichier définis dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue est composée de noms symboliques de fichier sur 8 caractères complétés avec des espaces si nécessaire.

3.2.3.2 Visualisation d'un fichier (*ApiCxNetViewFile*)

Description

Cette fonction permet à l'application de visualiser la définition d'un fichier défini au répertoire des fichiers du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *FileName* : le nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_FILE donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_FILE est complétée de la manière suivante :

- *Comment* : libellé décrivant la définition de ce fichier sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du fichier et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *Définition* : règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Direction* : direction autorisée de transfert de ce fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Réception et Transmission
- *FileType* : type du fichier et peut prendre les valeurs 'TF' pour Texte Fixe, 'TV' pour Texte Variable, 'BF' pour Binaire Fixe ou 'BI' pour Binaire Indéfini.
- *ReceiveMode* : règle en réception et peut prendre les valeurs 'APICX_RECVN' pour Nouveau ou 'APICX_RECVR' pour Remplacer. Pour compatibilité avec le sens Transmission, ce champ prend la valeur 'APICX_SENDO' pour Ouverture d'un fichier existant
- *Origin* : nom du partenaire autorisé à émettre ce fichier sur 8 caractères en majuscules, complété par des espaces
- *Destination* : nom du partenaire autorisé à recevoir ce fichier sur 8 caractères en majuscules, complété par des espaces
- *PresName* : nom de la table de présentation du protocole PeSIT utilisée pour ce fichier sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *PhysName* : nom physique de ce fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *RecordLg* : taille des enregistrements de ce fichier sur au plus 5 caractères, complété par des espaces

- *TrIex* : nom de l'exit de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTex* : nom de l'exit de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIex* : nom de l'exit de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTex* : nom de l'exit de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrIcd* : nom de la commande de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTcd* : nom de la commande de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIcd* : nom de la commande de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTcd* : nom de la commande de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Errcd* : nom de la commande d'erreur de transfert sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Notify* : permet de mettre en œuvre la notification de transfert pour ce fichier 'O' ou non 'N'
- *CliName* : le nom du client à notifier sur au plus 8 caractères en majuscules, complété par des espaces

3.2.3.3 Visualisation d'un fichier (Extension) (ApiCxNetViewFileEx)

Description

Cette fonction permet à l'application de visualiser la définition d'un fichier défini au répertoire des fichiers du moniteur connecté.

C'est une extension de la fonction ApiCxNetViewFile précédente permettant de traiter les informations relatives au pi 37 (label) et au pi 99. Voir le document *Utilisation des pi 37 et 99 avec un partenaire PeSIT*. La structure utilisée est la structure APICXN_FILE_EX. Les appels à cette fonction ne peuvent s'adresser qu'à des moniteurs Connect:Express dont la version est supérieure ou égale à V302.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *FileName* : le nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_FILE_EX donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le *Guide utilisateur de Connect:Express Windows*. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_FILE_EX est complétée de la manière suivante :

- *Comment* : libellé décrivant la définition de ce fichier sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du fichier et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *Definition* : règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Direction* : direction autorisée de transfert de ce fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Réception et Transmission
- *FileType* : type du fichier et peut prendre les valeurs 'TF' pour Texte Fixe, 'TV' pour Texte Variable, 'BF' pour Binaire Fixe ou 'BI' pour Binaire Indéfini.
- *ReceiveMode* : règle en réception et peut prendre les valeurs 'APICX_RECVN' pour Nouveau ou 'APICX_RECVR' pour Remplacer. Pour compatibilité avec le sens Transmission, ce champ prend la valeur 'APICX_SENDO' pour Ouverture d'un fichier existant
- *Origin* : nom du partenaire autorisé à émettre ce fichier sur 8 caractères en majuscules, complété par des espaces
- *Destination* : nom du partenaire autorisé à recevoir ce fichier sur 8 caractères en majuscules, complété par des espaces
- *PresName* : nom de la table de présentation du protocole PeSIT utilisée pour ce fichier sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *PhysName* : nom physique de ce fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *RecordLg* : taille des enregistrements de ce fichier sur au plus 5 caractères, complété par des espaces

- *TrIex* : nom de l'exit de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTex* : nom de l'exit de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIex* : nom de l'exit de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTex* : nom de l'exit de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrIcd* : nom de la commande de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTcd* : nom de la commande de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIcd* : nom de la commande de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTcd* : nom de la commande de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Errcd* : nom de la commande d'erreur de transfert sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Notify* : permet de mettre en œuvre la notification de transfert pour ce fichier 'O' ou non 'N'
- *CliName* : le nom du client à notifier sur au plus 8 caractères en majuscules, complété par des espaces
- *Pi99LoadTypeT* : Valeur 'D'
- *Pi99OffsetT* : Offset pi 99 en émission (3 caractères numériques)
- *Pi99LengthT* : Longueur pi 99 en émission (3 caractères numériques)
- *Pi9ValueT* : Valeur pi 99 en émission (254 caractères) (padding espace)
- *Pi99LoadTypeR* : Valeur 'D'
- *Pi99OffsetR* : Offset du pi 99 en réception (3 caractères numériques)
- *Pi99LengthR* : Longueur pi 99 en réception (3 caractères numériques)
- *Pi9ValueR* : Valeur pi 99 en réception (254 caractères) (padding espace)
- *Label* : Valeur du pi 37 (80 caractères) (padding espace)
- *TypeOfNotification* : Type de notification HTTP, utilisé par le moniteur uniquement si la composante de notification HTTP est installée. 1 caractère ('0' à '7'). '0' : pas de notification http. '1' : Notification http en début de transfert. '2' : Notification http en fin de transfert. '4' : Notification http si erreur de transfert. Les autres possibilités sont des combinaisons par « OU » inclusif de ces valeurs. Par exemple : '6' = '2' OR '4' pour une notification en fin de transfert ou en cas d'erreur de transfert.
- *Filler[255]* : réservé.

3.2.3.4 Ajout/Modification d'un fichier (ApiCxNetUpdateFile)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un fichier au répertoire des fichiers du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *FileName* : le nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces. Si le fichier n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *Comment* : libellé décrivant la définition de ce fichier sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du fichier et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *Définition* : règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Direction* : direction autorisée de transfert de ce fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Réception et Transmission
- *FileType* : type du fichier et peut prendre les valeurs 'TF' pour Texte Fixe, 'TV' pour Texte Variable, 'BF' pour Binaire Fixe ou 'BI' pour Binaire Indéfini.
- *ReceiveMode* : règle en réception et peut prendre les valeurs 'APICX_RECVN' pour Nouveau ou 'APICX_RECVR' pour Remplacer. Pour compatibilité avec le sens Transmission, ce champ prend la valeur 'APICX_SENDO' pour Ouverture d'un fichier existant
- *Origin* : nom du partenaire autorisé à émettre ce fichier sur 8 caractères en majuscules, complété par des espaces
- *Destination* : nom du partenaire autorisé à recevoir ce fichier sur 8 caractères en majuscules, complété par des espaces
- *PresName* : nom de la table de présentation du protocole PeSIT utilisée pour ce fichier sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *PhysName* : nom physique de ce fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *RecordLg* : taille des enregistrements de ce fichier sur au plus 5 caractères, complété par des espaces
- *TrLex* : nom de l'exit de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTex* : nom de l'exit de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReLex* : nom de l'exit de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTex* : nom de l'exit de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrIcd* : nom de la commande de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTcd* : nom de la commande de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIcd* : nom de la commande de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTcd* : nom de la commande de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Errcd* : nom de la commande d'erreur de transfert sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Notify* : permet de mettre en œuvre la notification de transfert pour ce fichier 'O' ou non 'N'
- *CliName* : le nom du client à notifier sur au plus 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_FILE donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.3.5 Ajout/Modification d'un fichier (Extension) (ApiCxNetUpdateFileEx)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un fichier au répertoire des fichiers du moniteur connecté.

C'est une extension de la fonction *ApiCxNetUpdateFile* précédente permettant de traiter les informations relatives au pi 37 (label) et au pi 99. Voir le document *Utilisation des pi 37 et 99 avec un partenaire PeSIT*. La structure utilisée est la structure APICXN_FILE_EX. Les appels à cette fonction ne peuvent s'adresser qu'à des moniteurs Connect:Express dont la version est supérieure ou égale à V302.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *FileName* : le nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces. Si le fichier n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *Comment* : libellé décrivant la définition de ce fichier sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du fichier et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *Definition* : règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Direction* : direction autorisée de transfert de ce fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Réception et Transmission
- *FileType* : type du fichier et peut prendre les valeurs 'TF' pour Texte Fixe, 'TV' pour Texte Variable, 'BF' pour Binaire Fixe ou 'BI' pour Binaire Indéfini.
- *ReceiveMode* : règle en réception et peut prendre les valeurs 'APICX_RECVN' pour Nouveau ou 'APICX_RECVR' pour Remplacer. Pour compatibilité avec le sens Transmission, ce champ prend la valeur 'APICX_SENDO' pour Ouverture d'un fichier existant
- *Origin* : nom du partenaire autorisé à émettre ce fichier sur 8 caractères en majuscules, complété par des espaces
- *Destination* : nom du partenaire autorisé à recevoir ce fichier sur 8 caractères en majuscules, complété par des espaces
- *PresName* : nom de la table de présentation du protocole PeSIT utilisée pour ce fichier sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *PhysName* : nom physique de ce fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *RecordLg* : taille des enregistrements de ce fichier sur au plus 5 caractères, complété par des espaces
- *TrIex* : nom de l'exit de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTex* : nom de l'exit de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)

- *ReIex* : nom de l'exit de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTex* : nom de l'exit de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrIcd* : nom de la commande de début de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *TrTcd* : nom de la commande de fin de transmission sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReIcd* : nom de la commande de début de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *ReTcd* : nom de la commande de fin de réception sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Errcd* : nom de la commande d'erreur de transfert sur au plus 127 caractères et terminé par un caractère nul (\0)
- *Notify* : permet de mettre en œuvre la notification de transfert pour ce fichier 'O' ou non 'N'
- *CliName* : le nom du client à notifier sur au plus 8 caractères en majuscules, complété par des espaces
- *Pi99LoadTypeT* : Valeur 'D'
- *Pi99OffsetT* : Offset pi 99 en émission (3 caractères numériques)
- *Pi99LengthT* : Longueur pi 99 en émission (3 caractères numériques)
- *Pi9ValueT* : Valeur pi 99 en émission (254 caractères) (padding espace)
- *Pi99LoadTypeR* : Valeur 'D'
- *Pi99OffsetR* : Offset du pi 99 en réception (3 caractères numériques)
- *Pi99LengthR* : Longueur pi 99 en réception (3 caractères numériques)
- *Pi9ValueR* : Valeur pi 99 en réception (254 caractères) (padding espace)
- *Label* : Valeur du pi 37 (80 caractères) (padding espace)
- *TypeOfNotification* : Type de notification HTTP, utilisé par le moniteur uniquement si la composante de notification HTTP est installée. 1 caractère ('0' à '7'). '0' : pas de notification http. '1' : Notification http en début de transfert. '2' : Notification http en fin de transfert. '4' : Notification http si erreur de transfert. Les autres possibilités sont des combinaisons par « OU » inclusif de ces valeurs. Par exemple : '6' = '2' OR '4' pour une notification en fin de transfert ou en cas d'erreur de transfert.
- *Filler[255]* : réservé.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_FILE_EX donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.3.6 Suppression d'un fichier (*ApiCxNetDelFile*)

Description

Cette fonction permet à l'application de supprimer la définition d'un fichier au répertoire des fichiers du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *FileName* : le nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_FILE donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.4 Accès au répertoire des clients

3.2.4.1 Liste des clients (*ApiCxNetListClient*)

Description

Cette fonction permet à l'application d'obtenir la liste des clients définis au répertoire des clients du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des clients
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXN_LIST_CLI* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXN_LIST_CLI* est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms symboliques de client dans la liste
- *TotDef* : nombre de noms symboliques de client définis dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue est composée de noms symboliques de client sur 8 caractères complétés avec des espaces si nécessaire.

3.2.4.2 Visualisation d'un client (*ApiCxNetViewClient*)

Description

Cette fonction permet à l'application de visualiser la définition d'un client défini au répertoire des clients du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : le nom symbolique du client sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_CLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_CLI est complétée de la manière suivante :

- *CliPsw* : mot de passe du client sur 8 caractères en majuscules, complété par des espaces
- *Comment* : libellé décrivant la définition de ce client sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du client et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *CliAuth* : liste des droits du client (voir APICXV3D.H structure APICXN_CX)
- *FilesNb* : nombre de fichiers autorisés pour le transfert par ce client sur 3 caractères numériques de '000' à '100'
- *Files* : liste des noms symboliques de fichiers autorisés à être transférés par ce client sur 8 caractères en majuscules et complété par des espaces
- *Notify* : permet à ce client de recevoir des notifications de transfert 'O' ou non 'N'
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce client et peut prendre les valeurs 'APICX_TCPIP' ou 'APICX_NP'
- *IpAddr* : adresse IP du client au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du client sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *IpPort* : numéro de port TCP/IP du client sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *NpName* : nom du Canal Nommé du client sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.4.3 Ajout/Modification d'un client (*ApiCxNetUpdateClient*)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un client au répertoire des clients du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : le nom symbolique du client sur 8 caractères en majuscules, complété par des espaces. Si le client n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *CliPsw* : mot de passe du client sur 8 caractères en majuscules, complété par des espaces
- *Comment* : libellé décrivant la définition de ce client sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du client et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *CliAuth* : liste des droits du client (voir APICXV3D.H structure APICXN_CX)
- *FilesNb* : nombre de fichiers autorisés pour le transfert par ce client sur 3 caractères numériques de '000' à '100'
- *Files* : liste des noms symboliques de fichiers autorisés à être transférés par ce client sur 8 caractères en majuscules et complété par des espaces
- *Notify* : permet à ce client de recevoir des notifications de transfert 'O' ou non 'N'
- *Linktp* : type de réseau utilisé par défaut pour communiquer avec ce client et peut prendre les valeurs 'APICX_TCPIP' ou 'APICX_NP'
- *IpAddr* : adresse IP du client au format 'aaa.bbb.ccc.ddd' et terminée par au moins un caractère nul (\0)
- *IpName* : nom de 'host' TCP/IP du client sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *IpPort* : numéro de port TCP/IP du client sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *NpName* : nom du Canal Nommé du client sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_CLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.4.4 Suppression d'un client (*ApiCxNetDelClient*)

Description

Cette fonction permet à l'application de supprimer la définition d'un client au répertoire des clients du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : le nom symbolique du client sur 8 caractères en majuscules, complété par des espaces.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_CLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.5 Accès au répertoire des paramètres des clients SSL

3.2.5.1 Liste des clients SSL (*ApiCxNetListSslparmCli*)

Description

Cette fonction permet à l'application d'obtenir la liste des paramètres de clients SSL définis au répertoire des clients SSL du moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des clients SSL
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_SSLPARAMCLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_SSLPARAMCLI est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms symboliques de clients SSL dans la liste
- *TotDef* : nombre de noms symboliques de clients SSL définis dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue est composée de noms symboliques de clients SSL sur 8 caractères complétés avec des espaces si nécessaire.

3.2.5.2 Visualisation d'un client SSL (*ApiCxNetViewSslparmCli*)

Description

Cette fonction permet à l'application de visualiser la définition des paramètres d'un client SSL défini au répertoire des clients SSL du moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *Name* : le nom symbolique du client SSL sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_SSLLPARAMCLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_SSLLPARAMCLI est complétée de la manière suivante :

- *Enabled* : état du client ('1' : En service, '0' : Hors service)
- *StoreProvider* : Fournisseur de certificat. Chaîne de caractères 'STORE_PROV_SYSTEM' terminée par un zéro binaire.
- *StoreLocation* : Emplacement du certificat. Chaîne de caractères terminée par un zéro binaire pouvant prendre l'une des valeurs : 'SYSTEM_STORE_LOCAL_MACHINE', 'SYSTEM_STORE_SERVICES' ou 'SYSTEM_STORE_CURRENT_USER'.
- *StoreName* : Chaîne de caractères 'My' terminée par un zéro en cas d'authentification du client. Champ facultatif.
- *SubjectDn* : Distinguished name objet du certificat du client en cas d'authentification de celui-ci. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *IssuerDn* : Distinguished name émetteur du certificat du client en cas d'authentification de celui-ci. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *Protocol* : Version de protocole SSL utilisée ('1' : TLSV1, '3' : SSLV3, '2' : SSLV2)
- *CipherSuites* : Suites de chiffrement. Chaîne éventuellement vide formée d'une liste de n° de 2 digits chacun, indiquant les suites de chiffrement à utiliser. La chaîne ainsi formée est terminée par un zéro binaire. Les numéros suivant peuvent être utilisés :
 - '00' TLS_RSA_WITH_RC4_128_MD5
 - '01' TLS_RSA_WITH_RC4_128_SHA
 - '02' TLS_RSA_WITH_3DES_EDE_CBC_SHA,
 - '03' TLS_RSA_WITH_DES_CBC_SHA,
 - '06' SSLCK_RC4_128_WITH_MD5,
 - '07' SSLCK_DES_64_CBC_WITH_MD5,
 - '08' SSLCK_RC2_128_CBC_WITH_MD5Exemple : 000103\0 indique les suites 00, 01 et 03.
- *Trace* : Indicateur de trace ('0' : Aucune, '1' : Trace partielle, '2' : Trace complète)

3.2.5.3 Visualisation d'un client SSL (ApiCxNetViewSslparamCliEx)

Description

Cette fonction est identique à la précédente, mais la structure rendue contient en plus le champ :

- *SslBytesHeader* : Indique que les données non encryptées sont préfixées avec 2 octets de longueur ('0' : Pas d'octets de longueur, '1' : Il y a 2 octets de longueur)

La structure est nommée APICXN_SSLLPARAMCLI_EX.

3.2.5.4 Ajout/Modification d'un client SSL(ApiCxNetUpdateSslparamCli)

Description

Cette fonction permet à l'application d'ajouter ou de modifier la définition d'un client SSL au répertoire des clients SSL du moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *Name* : le nom symbolique du client SSL sur 8 caractères en majuscules, complété par des espaces. Si le client SSL n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *Enabled* : état du client ('1' : En service, '0' : Hors service)
- *StoreProvider* : Fournisseur de certificat. Chaîne de caractères 'STORE_PROV_SYSTEM' terminée par un zéro binaire.
- *StoreLocation* : Emplacement du certificat. Chaîne de caractères terminée par un zéro binaire pouvant prendre l'une des valeurs : 'SYSTEM_STORE_LOCAL_MACHINE', 'SYSTEM_STORE_SERVICES' ou 'SYSTEM_STORE_CURRENT_USER'.
- *StoreName* : Chaîne de caractères 'My' terminée par un zéro binaire à renseigner en cas d'authentification du client. Champ facultatif.
- *SubjectDn* : Distinguished name objet du certificat du client à renseigner en cas d'authentification client. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *IssuerDn* : Distinguished name émetteur du certificat du client à renseigner en cas d'authentification client. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *Protocol* : Version de protocole SSL utilisée ('1' : TLSV1, '3' : SSLV3, '2' : SSLV2)
- *CipherSuites* : Suites de chiffrement. Renseigner une chaîne éventuellement vide formée d'une liste de n° de 2 digits chacun, indiquant les suites de chiffrement à utiliser. La chaîne ainsi formée doit être terminée par un zéro binaire. Les numéros suivant peuvent être utilisés :
 - '00' TLS_RSA_WITH_RC4_128_MD5
 - '01' TLS_RSA_WITH_RC4_128_SHA
 - '02' TLS_RSA_WITH_3DES_EDE_CBC_SHA,
 - '03' TLS_RSA_WITH_DES_CBC_SHA,
 - '06' SSL CK_RC4_128_WITH_MD5,
 - '07' SSL CK_DES_64_CBC_WITH_MD5,
 - '08' SSL CK_RC2_128_CBC_WITH_MD5Exemple : 000103\0 indique les suites 00, 01 et 03.
- *Trace* : Indicateur de trace ('0' : Aucune, '1' : Trace partielle, '2' : Trace complète)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_SSLPARMCLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.5.5 Ajout/Modification d'un client SSL(ApiCxNetUpdateSslparmCliEx)

Description

Cette fonction est identique à la précédente, mais la structure fournie contient en plus le champ :

- *SslBytesHeader* : Indique que les données non encryptées sont préfixées avec 2 octets de longueur ('0' : Pas d'octets de longueur, '1' : Il y a 2 octets de longueur)

La structure est nommée APICXN_SSLPARMCLI_EX.

3.2.5.6 Suppression d'un client SSL (*ApiCxNetDelSslparmCli*)

Description

Cette fonction permet à l'application de supprimer la définition d'un client SSL du répertoire des clients SSL du moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *Name* : le nom symbolique du client SSL à supprimer sur 8 caractères en majuscules, complété par des espaces.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_SSLPARAMCLI donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.6 Accès à l'activité du moniteur de transferts

3.2.6.1 Liste de noms autorisés (*ApiCxNetListAuth*)

Description

Cette fonction permet à l'application d'obtenir la liste des fichiers autorisés pour un client donné ou la liste des partenaires autorisés pour un fichier donné.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListTyp* : le type de liste qui peut être APICX_FILELIST, APICX_TRANLIST ou APICX_RECELIST.
- *CliName* : le nom symbolique du client pour la liste des fichiers autorisés
- *FileName* : le nom symbolique du fichier pour la liste des partenaires autorisés

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_AUTH donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_AUTH est complétée de la manière suivante :

- *NamesNb* : nombre de noms symboliques dans la table
- *Names* : table des noms

La liste obtenue est composée de noms symboliques sur 8 caractères complétés avec des espaces si nécessaire.

3.2.6.2 Détail d'un fichier autorisé (*ApiCxNetViewAuth*)

Description

Cette fonction permet à l'application d'obtenir le détail de la définition d'un fichier autorisé pour un client donné.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : le nom symbolique du client
- *FileName* : le nom symbolique du fichier autorisé pour le client

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DETAIL_AUTH donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_DETAIL_AUTH est complétée de la manière suivante :

- *Comment* : libellé décrivant la définition de ce fichier sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *State* : état du fichier et peut prendre les valeurs 'APICX_STAE' pour En service ou 'APICX_STAH' pour Hors service
- *Definition* : règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Direction* : direction autorisée de transfert de ce fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Réception et Transmission
- *Origin* : nom du partenaire autorisé à émettre ce fichier sur 8 caractères en majuscules, complété par des espaces
- *Destination* : nom du partenaire autorisé à recevoir ce fichier sur 8 caractères en majuscules, complété par des espaces
- *PhysName* : nom physique de ce fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.6.3 Type de définition pour un partenaire (*ApiCxNetPartAuth*)

Description

Cette fonction permet à l'application de savoir si l'identification locale pour un partenaire donné est statique ou dynamique.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PartName* : le nom symbolique du partenaire

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PART_AUTH donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PART_AUTH est complétée de la manière suivante :

- *LocTyp* : indique si l'identifiant local est statique (APICX_LOCS) ou dynamique (APICX_LOCD).

3.2.6.4 Soumission d'une requête de transfert (ApiCxNetSubmitReq)

Description

Cette fonction permet à l'application de soumettre une requête de transfert au moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client sur 8 caractères en majuscules, complété par des espaces. C'est le client qui soumet la requête. Le client connecté est pris par défaut.
- *FileName* : nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces
- *PhysName* : nom physique du fichier sur au plus 127 caractères et terminé par au moins un caractère nul (0). Le nom physique défini au répertoire des fichiers est utilisé par défaut
- *Direction* : direction du transfert et peut prendre les valeurs 'APICX_DIRR' pour Réception ou 'APICX_DIRT' pour Transmission ou espace pour prendre la valeur par défaut. La direction définie au répertoire est prise par défaut
- *PartName* : nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. En fonction de la direction, le nom de l'émetteur ou du destinataire défini au répertoire des fichiers est utilisé par défaut
- *Linktp* : type de réseau utilisé pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62' ou espace pour prendre la valeur par défaut. Le type de réseau défini au répertoire des partenaires est pris par défaut
- *Alias* : nom symbolique local quand le partenaire indique une identification locale dynamique
- *AliasPsw* : mot de passe local quand le partenaire indique une identification locale dynamique
- *Priority* : priorité du transfert et peut prendre les valeurs 'APICX_PRIL' pour Lente, 'APICX_PRIN' pour Normale ou 'APICX_PRIU' pour Urgente ou espace pour prendre la valeur par défaut. La priorité est Normale par défaut.
- *ReqType* : type de la requête et peut prendre les valeurs 'APICX_RTYPD' pour Demande (Inquiry), 'APICX_RTYPN' pour Normale ou 'APICX_RTYPA' pour Attente (hold) ou espace pour prendre la valeur par défaut. Le type de la requête est Normale pour un transfert en Transmission et Demande pour un transfert en Réception
- *CliIdent* : identifiant client de ce transfert sur 16 caractères libres
- *Notify* : permet de mettre en œuvre la notification pour ce transfert.
- *NotifyInfos* : structure contenant uniquement, dans cette version, le nom du client à notifier sur au plus 8 caractères, complété par des espaces
- *AdHoc* : structure utilisée dans le cadre du libre-service et contenant :
 - le nom de l'utilisateur sur 8 caractères
 - le mot de passe de l'utilisateur sur 8 caractères
 - le nom physique du fichier local sur 44 caractères
 - le nom physique du fichier distant sur 44 caractères
- *Extend* : structure contenant les informations étendues de la requête et contenant :
 - l'origine du fichier sur 8 caractères
 - la destination du fichier sur 8 caractères
 - l'émetteur du fichier sur 8 caractères
 - le récepteur du fichier sur 8 caractères
 - le label du fichier sur 80 caractères
- *TrfDate* : date d'exécution du transfert au format AAAA/MM/JJ ou espace pour prendre la valeur par défaut. La date du jour est prise par défaut
- *TrfTime* : heure d'exécution du transfert au format HH :MM :SS ou espace pour prendre la valeur par défaut. L'heure courante est prise par défaut
- *Etebac-3* : carte ETEBAC-3

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_SUB_REQ donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_SUB_REQ est complétée de la manière suivante :

- *ReqNb* : indiqué par l'API , c'est le nombre de requêtes de transfert soumises au moniteur sur 3 caractères numériques de '000' à '100'
- *ReqNumber* : liste des numéros de requête de transfert sur 12 caractères en majuscules et complété par des espaces

3.2.6.5 Soumission d'une requête de transfert (Extension) (ApiCxNetSubmitReqEx)

Description

Cette fonction permet à l'application de soumettre une requête de transfert au moniteur connecté.

C'est une extension de la fonction *ApiCxNetSubmitReq* précédente permettant de traiter les informations relatives au au pi 99. Voir le document *Utilisation des pi 37 et 99 avec un partenaire PeSIT*. La structure utilisée est la structure APICXN_SUB_REQ_EX. Les appels à cette fonction ne peuvent s'adresser qu'à des moniteurs Connect:Express dont la version est supérieure ou égale à V302.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client sur 8 caractères en majuscules, complété par des espaces. C'est le client qui soumet la requête. Le client connecté est pris par défaut.
- *FileName* : nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces
- *PhysName* : nom physique du fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0). Le nom physique défini au répertoire des fichiers est utilisé par défaut
- *Direction* : direction du transfert et peut prendre les valeurs 'APICX_DIRR' pour Réception ou 'APICX_DIRT' pour Transmission ou espace pour prendre la valeur par défaut. La direction définie au répertoire est prise par défaut
- *PartName* : nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. En fonction de la direction, le nom de l'émetteur ou du destinataire défini au répertoire des fichiers est utilisé par défaut
- *Linktp* : type de réseau utilisé pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP', 'APICX_X25' ou 'APICX_LU62' ou espace pour prendre la valeur par défaut. Le type de réseau défini au répertoire des partenaires est pris par défaut
- *Alias* : nom symbolique local quand le partenaire indique une identification locale dynamique
- *AliasPsw* : mot de passe local quand le partenaire indique une identification locale dynamique
- *Priority* : priorité du transfert et peut prendre les valeurs 'APICX_PRIL' pour Lente, 'APICX_PRIN' pour Normale ou 'APICX_PRIU' pour Urgente ou espace pour prendre la valeur par défaut. La priorité est Normale par défaut.
- *ReqType* : type de la requête et peut prendre les valeurs 'APICX_RTYPD' pour Demande (Inquiry), 'APICX_RTYPN' pour Normale ou 'APICX_RTYPA' pour Attente (hold) ou espace pour prendre la valeur par défaut. Le type de la requête est Normale pour un transfert en Transmission et Demande pour un transfert en Réception
- *CliIdent* : identifiant client de ce transfert sur 16 caractères libres

- *Notify* : permet de mettre en œuvre la notification pour ce transfert.
- *NotifyInfos* : structure contenant uniquement, dans cette version, le nom du client à notifier sur au plus 8 caractères, complété par des espaces
- *AdHoc* : structure utilisée dans le cadre du libre-service et contenant :
 - le nom de l'utilisateur sur 8 caractères
 - le mot de passe de l'utilisateur sur 8 caractères
 - le nom physique du fichier local sur 44 caractères
 - le nom physique du fichier distant sur 44 caractères
- *Extend* : structure contenant les informations étendues de la requête et contenant :
 - l'origine du fichier sur 8 caractères
 - la destination du fichier sur 8 caractères
 - l'émetteur du fichier sur 8 caractères
 - le récepteur du fichier sur 8 caractères
 - le label du fichier sur 80 caractères
- *TrfDate* : date d'exécution du transfert au format AAAA/MM/JJ ou espace pour prendre la valeur par défaut. La date du jour est prise par défaut
- *TrfTime* : heure d'exécution du transfert au format HH :MM :SS ou espace pour prendre la valeur par défaut. L'heure courante est prise par défaut
- *Etebac-3* : carte ETEBAC-3
- *Pi99LoadTyp* : Valeur 'D'
- *Pi99Offset* : Offset pi 99 (3 caractères numériques)
- *Pi99Length* : Longueur pi 99 (3 caractères numériques)
- *Pi9Value* : Valeur pi 99 (254 caractères) (padding espace)
- *TypeOfNotification* : Type de notification HTTP, utilisé par le moniteur uniquement si la composante de notification HTTP est installée. 1 caractère ('0' à '7'). '0' : pas de notification http. '1' : Notification http en début de transfert. '2' : Notification http en fin de transfert. '4' : Notification http si erreur de transfert. Les autres possibilités sont des combinaisons par « OU » inclusif de ces valeurs. Par exemple : '6' = '2' OR '4' pour une notification en fin de transfert ou en cas d'erreur de transfert.
- *Filler[255]* : réservé.

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_SUB_REQ_EX donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_SUB_REQ_EX est complétée de la manière suivante :

- *ReqNb* : indiqué par l'API , c'est le nombre de requêtes de transfert soumises au moniteur sur 3 caractères numériques de '000' à '100'
- *ReqNumber* : liste des numéros de requête de transfert sur 12 caractères en majuscules et complété par des espaces

3.2.6.6 Liste des requêtes de transfert (*ApiCxNetListReq*)

Description

Cette fonction permet à l'application d'obtenir la liste des requêtes du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ReqUserType* : type de l'initiateur de la requête qui peut prendre les valeurs 'APICX_USRTTI' pour interne, 'APICX_USRTE' pour externe ou 'APICX_USR' pour interne et externe.
- *CliName* : nom symbolique du client initiateur de la requête sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les clients
- *ClIdent* : identifiant client de ce transfert sur 16 caractères libres. La valeur '*' permet de sélectionner tous les identifiants.
- *FileName* : nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les fichiers
- *Direction* : direction du transfert du fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Transmission ou Réception
- *PartName* : nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les partenaires
- *ListAddr* : adresse d'un emplacement mémoire alloué par l'application qui va recevoir la liste des requêtes répondant aux critères de sélection indiqués ci avant
- *ListSize* : taille en octets de l'emplacement mémoire

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_REQ donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_REQ est complétée de la manière suivante :

- *NbReqRet* : nombre de requêtes mémorisées dans la liste indiquée en entrée
- *TotReq* : nombre total de requêtes répondant aux critères de sélection

Si le nombre total de requêtes est supérieur au nombre de requêtes mémorisées dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue constitue une table de structures APICXN_DLIST_REQ dont chaque poste est composé de la manière suivante :

- *ReqNumber* : numéro de requête de transfert sur 12 caractères
- *CliIdent* : identifiant client du transfert sur 16 caractères
- *CliName* : nom du client initiateur de la requête de transfert sur 8 caractères
- *FileName* : nom symbolique du fichier sur 8 caractères
- *Direction* : direction du transfert et peut prendre les valeurs 'APICX_DIRT' ou 'APICX_DIRR'
- *PartName* : nom symbolique du partenaire sur 8 caractères
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours
- *ReqUserType* : type de l'initiateur de la requête qui peut prendre les valeurs 'APICX_USRTI' pour interne, 'APICX_USRTE' pour externe

3.2.6.7 Visualisation d'une requête de transfert (*ApiCxNetViewReq*)

Description

Cette fonction permet à l'application d'obtenir le détail d'une requête de transfert traitée par le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ReqNumber* : numéro de requête de transfert

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_REQ donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_REQ est complétée de la manière suivante :

- *CliIdent* : identifiant client de ce transfert sur 16 caractères libres
- *CliName* : nom symbolique du client initiateur de la requête sur 8 caractères
- *FileName* : nom symbolique du fichier sur 8 caractères
- *Direction* : direction du transfert du fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception ou 'APICX_DIRT' pour Transmission
- *PartName* : nom symbolique du partenaire sur 8 caractères
- *PhysName* : nom physique du fichier sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours
- *ReqUserType* : type de l'initiateur de la requête qui peut prendre les valeurs 'APICX_USRTI' pour interne ou 'APICX_USRTE' pour externe
- *Trc* : code Connect:Express de l'erreur
- *Prc* : code Protocole de l'erreur
- *Src* : code Système de l'erreur
- *Erc* : code C-TREE de l'erreur
- *Nrc* : code Réseau de l'erreur
- *FileSize* : taille du fichier en octets
- *TransSize* : taille transférée du fichier en octets
- *BegDate* : date de début du transfert au format AAA/MM/JJ
- *BegTime* : heure de début du transfert au format HH :MM :SS

3.2.6.8 Action sur une requête de transfert (*ApiCxNetActionReq*)

Description

Cette fonction permet à l'application d'agir sur une requête de transfert du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ReqNumber* : numéro de requête de transfert
- *Action* action à effectuer sur la requête et peut prendre les valeurs 'APICX_ACTI' pour Interrompre, 'APICX_ACTP' pour Purger ou 'APICX_ACTR' pour Reprendre

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_ACTION_REQ donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.7 Accès aux messages du moniteur de transferts

3.2.7.1 Liste des messages (*ApiCxNetListLog*)

Description

Cette fonction permet à l'application d'obtenir la liste des messages du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *BegDate* : date de début des messages au format AAAA/MM/JJ
- *BegTime* : heure de début des messages au format HH :MM :SS
- *EndDate* : date de fin des messages au format AAAA/MM/JJ
- *EndTime* : heure de fin des messages au format HH :MM :SS
- *ListAddr* : adresse d'un emplacement mémoire alloué par l'application qui va recevoir la liste des messages du moniteur compris dans l'intervalle de temps indiqué ci avant
- *ListSize* : taille en octets de l'emplacement mémoire

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_LOG donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_LOG est complétée de la manière suivante :

- *NbMsgRet* : nombre de messages mémorisées dans la liste indiquée en entrée
- *TotMsg* : nombre total de messages répondant aux critères de sélection

Si le nombre total de messages est supérieur au nombre de messages mémorisés dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste retournée présente une suite de messages de 255 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.2.8 Accès au journal des transferts du moniteur de transferts

3.2.7.1 Liste du journal (*ApiCxNetListJnl*)

Description

Cette fonction permet à l'application d'obtenir la liste des requêtes mémorisées dans le journal des transferts du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client initiateur de la requête sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les clients
- *FileName* : nom symbolique du fichier sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les fichiers
- *PartName* : nom symbolique du partenaire sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les partenaires
- *Direction* : direction du transfert du fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception, 'APICX_DIRT' pour Transmission ou 'APICX_DIR' pour Transmission ou Réception
- *BegDate* : date de début des enregistrements journal
- *BegTime* : heure de début des enregistrements journal
- *EndDate* : date de fin des enregistrements journal
- *EndTime* : heure de fin des enregistrements journal
- *ListAddr* : adresse d'un emplacement mémoire alloué par l'application qui va recevoir la liste des enregistrements journal répondant aux critères de sélection indiqués ci avant
- *ListSize* : taille en octets de l'emplacement mémoire

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_JNL donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_JNL est complétée de la manière suivante :

- *NbJnlRet* : nombre d'enregistrements journal mémorisés dans la liste indiquée en entrée
- *TotJnl* : nombre total d'enregistrements journal répondant aux critères de sélection

Si le nombre total d'enregistrements journal est supérieur au nombre d'enregistrements journal mémorisés, cela indique que la taille de l'emplacement mémoire spécifié en entrée n'est pas assez grand pour permettre le stockage de la liste complète d'enregistrements journal

La liste obtenue constitue une table de structures APICXN_DLIST_JNL dont chaque poste est composé de la manière suivante :

- *JnlDate* : date d'inscription de l'enregistrement au journal au format AAAA/MM/JJ
- *JnlTime* : heure d'inscription de l'enregistrement au journal au format HH :MM :SS
- *ReqNumber* : numéro de requête de transfert sur 12 caractères
- *CliIdent* : identifiant client du transfert sur 16 caractères
- *CliName* : nom du client initiateur de la requête de transfert sur 8 caractères
- *FileName* : nom symbolique du fichier sur 8 caractères
- *Direction* : direction du transfert et peut prendre les valeurs 'APICX_DIRT' ou 'APICX_DIRR'
- *PartName* : nom symbolique du partenaire sur 8 caractères
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours

3.2.8.2 Visualisation d'un enregistrement du journal (*ApiCxNetViewJnl*)

Description

Cette fonction permet à l'application d'obtenir l'ensemble des informations liées à un transfert mémorisé dans le fichier journal du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ReqNumber* : numéro de requête de transfert

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_ JNL donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_ JNL est complétée de la manière suivante :

- *JnlDate* : date d'inscription de l'enregistrement au journal au format AAAA/MM/JJ
- *JnlTime* : heure d'inscription de l'enregistrement au journal au format HH :MM :SS
- *ExtReqNb* : numéro de requête du transfert chez le partenaire sur 8 caractères
- *CliIdent* : identifiant client de ce transfert
- *TrfId* : identifiant PeSIT du transfert
- *CliName* : nom symbolique du client initiateur de la requête
- *Trc* : code Connect:Express de l'erreur
- *Prc* : code Protocole de l'erreur
- *Src* : code Système de l'erreur
- *Erc* : code C-TREE de l'erreur
- *Nrc* : code Réseau de l'erreur
- *TcpipRc* : code TCP/IP de l'erreur
- *AppcPrc* : code primaire LU6.2 de l'erreur
- *AppcSrc* : code secondaire LU6.2 de l'erreur
- *X25rc* : code X25 de l'erreur
- *X25cause* : code CAUSE X25 de l'erreur
- *X25diag* : code DIAGNOSTIC X25 de l'erreur
- *ReqState* prend la valeur 'O' si la requête est purgée par le moniteur ou 'N' sinon
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours
- *ReqUserType* : type de l'initiateur de la requête qui peut prendre les valeurs 'APICX_USRTI' pour interne ou 'APICX_USRTE' pour externe
- *ReqType* : type de la requête et peut prendre les valeurs 'APICX_RTYPD' pour Demande (Inquiry), 'APICX_RTYPN' pour Normale ou 'APICX_RTYPA' pour Attente (hold)
- *Direction* : direction autorisée de transfert du fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception ou 'APICX_DIRT' pour Transmission
- *Priority* : priorité du transfert et peut prendre les valeurs 'APICX_PRIL' pour Lente, 'APICX_PRIN' pour Normale ou 'APICX_PRIU' pour Urgente
- *Linktp* : type de réseau utilisé pour communiquer avec ce partenaire et peut prendre les valeurs 'APICX_TCPIP' , 'APICX_X25' ou 'APICX_LU62'

- *FileName* : nom symbolique du fichier
- *PhysName* : nom physique du fichier
- *PhysOname* : nom physique d'origine du fichier
- *RemPhysName* : nom physique du fichier chez le partenaire
- *Label* : label du fichier
- *CreDateTime* : date et l'heure de création du fichier au format AAAAMMJJHHMMSS
- *MajDateTime* : date et l'heure de dernière mise à jour du fichier au format AAAAMMJJHHMMSS
- *Origin* : origine du fichier
- *Destination* : destination du fichier
- *Transmitter* : émetteur du fichier
- *Receiver* : récepteur du fichier
- *FileOrg* : organisation du fichier et peut prendre les valeurs 'APICX_ORGI' pour Indexée, 'APICX_ORGS' pour Séquentielle ou 'APICX_ORGR' pour Relative
- *FileType* : type du fichier et peut prendre les valeurs 'TF' pour Texte Fixe, 'TV' pour Texte Variable, 'BF' pour Binaire Fixe ou 'BI' pour Binaire Indéfini.
- *Definition* règle de définition du fichier et peut prendre les valeurs 'APICX_DEFF' pour Fixe ou 'APICX_DEFD' pour Dynamique
- *Compress* : type de compression demandé et peut prendre les valeurs 'APICX_NOCOMP' pour Pas de compression, 'APICX_HCOMP' pour Horizontale, 'APICX_VCOMP' pour Verticale ou 'APICX_MCOMP' pour Mixte
- *RealComp* : type de compression effectué et peut prendre les valeurs 'APICX_NOCOMP' pour Pas de compression, 'APICX_HCOMP' pour Horizontale, 'APICX_VCOMP' pour Verticale ou 'APICX_MCOMP' pour Mixte
- *DataType* : type de données contenues dans le fichier et peut prendre les valeurs 'APICX_DATA' pour Ascii, 'APICX_DATE' pour EBCDIC ou 'APICX_DATB' pour Binaire
- *RecordLg* : taille des enregistrements du fichier en octets
- *FileSize* : taille du fichier en octets
- *MsgSize* : taille des messages réseau en octets
- *LocName* : nom symbolique local de Connect:Express
- *PartName* : nom symbolique du partenaire
- *PartType* : type du partenaire et peut prendre les valeurs 'APICX_PCE' pour Connect:Express ou 'APICX_POTH' pour autre
- *ProtType* : type de protocole utilisé avec ce partenaire et peut prendre la valeur 'APICX_PPD' pour PeSIT-D, 'APICX_PPE' pour PeSIT-E ou 'APICX_PET3' pour ETEBAC-3
- *CrcOpt* : prend la valeur 'O' si le mécanisme de contrôle de CRC a été utilisé ou 'N' sinon
- *Sync* : intervalle de synchronisation en Kilo octets
- *Wind*: fenêtre de synchronisation
- *TcpipPort* : numéro de port TCP/IP du partenaire
- *TcpipAddr* : adresse IP du partenaire au format 'aaa.bbb.ccc.ddd'
- *TcpipName* : nom de 'host' TCP/IP du partenaire
- *LuName* : nom de la LU du partenaire
- *ModName* : nom du mode du partenaire
- *TpName* : nom du programme de transaction du partenaire
- *XRAddr* : adresse X.25 du partenaire
- *XLAddr* : adresse locale X.25 spécifiée lors de l'appel du partenaire
- *XnPort* : numéro de port X.25 utilisé pour ce partenaire
- *Xudata* : champ de données utilisateur spécifié lors de l'appel du partenaire
- *Xfac* : champ de facilités spécifié lors de l'appel du partenaire

- *BegDate* date de début du transfert au format AAA/MM/JJ
- *BegTime* : heure de début du transfert au format HH :MM :SS
- *EndDate* : date de fin du transfert au format AAA/MM/JJ
- *EndTime* heure de fin du transfert au format HH :MM :SS
- *RestartCnt* nombre de reprises effectuées pour ce transfert
- *OpenAct* : action effectuée à l'ouverture du fichier transféré et peut prendre les valeurs 'APICX_RECVN' pour Nouveau fichier créé, 'APICX_RECVR' pour fichier existant remplacé ou 'APICX_SENDO' pour fichier existant ouvert en lecture
- *Resync* : nombre de re-synchronisations effectuées pour ce transfert
- *TotByte* : nombre total d'octets pour ce transfert
- *TotRec* : nombre d'enregistrements transférés
- *TotIO* nombre d'entrées / sorties réseau effectuées pour ce transfert
- *ScheDate* : date de début de transfert demandée au format AAA/MM/JJ
- *ScheTime* : heure de début de transfert demandée au format HH :MM :SS

3.2.9 Accès aux notifications du moniteur de transferts

3.2.9.1 Liste des notifications (*ApiCxNetListNot*)

Description

Cette fonction permet à l'application d'obtenir la liste des notifications mémorisées dans le fichier des notifications du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client notifié sur 8 caractères en majuscules, complété par des espaces. La valeur '*' permet de sélectionner tous les clients
- *ListAddr* : adresse d'un emplacement mémoire alloué par l'application qui va recevoir la liste des notifications du client spécifié
- *ListSize* : taille en octets de l'emplacement mémoire

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_NOT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_NOT est complétée de la manière suivante :

- *NbNotRet* : nombre de notifications mémorisées dans la liste indiquée en entrée
- *TotNot* : nombre total de notifications répondant aux critères de sélection

Si le nombre total de notifications est supérieur au nombre de notifications mémorisées, cela indique que la taille de l'emplacement mémoire spécifié en entrée n'est pas assez grand pour permettre le stockage de la liste complète des notifications

La liste obtenue constitue une table de structures APICXN_DLIST_NOT dont chaque poste est composé de la manière suivante :

- *NotDate* : date de notification au format AAAA/MM/JJ
- *NotTime* : heure de notification au format HH :MM :SS
- *ReqNumber* : numéro de requête de transfert sur 12 caractères
- *CliIdent* : identifiant client du transfert sur 16 caractères
- *CliName* : nom du client initiateur de la requête de transfert sur 8 caractères
- *FileName* : nom symbolique du fichier sur 8 caractères
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours
- *PhysName* : nom physique du fichier
- *ReadInd* : indicateur positionné à 'O' (Oui) si la notification a déjà été lue, 'N' sinon.

3.2.9.2 Visualisation d'une notification (*ApiCxNetViewNot*)

Description

Cette fonction permet à l'application d'obtenir l'ensemble des informations d'une notification mémorisé dans le fichier des notifications du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client notifié
- *ReqNumber* : numéro de requête de transfert

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_NOT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_NOT est complétée de la manière suivante :

- *NotDate* : date de notification au format AAAA/MM/JJ
- *NotTime* : heure de notification au format HH :MM :SS
- *CliIdent* : identifiant client de ce transfert
- *ReqStatus* : état du transfert et peut prendre les valeurs 'APICX_REQSE' pour transfert terminé, 'APICX_REQSS' pour erreur de sélection, 'APICX_REQSI' pour Interrompu, 'APICX_REQSC' pour transfert en cours, 'APICX_REQST' pour sélection, 'APICX_REQSW' pour transfert en attente de sélection ou 'APICX_REQSR' pour reprise automatique en cours
- *Trc* : code Connect:Express de l'erreur
- *Prc* : code Protocole de l'erreur
- *Src* : code Système de l'erreur
- *Erc* : code C-TREE de l'erreur
- *Nrc* : code Réseau de l'erreur
- *Direction* : direction autorisée de transfert du fichier et peut prendre les valeurs 'APICX_DIRR' pour Réception ou 'APICX_DIRT' pour Transmission
- *FileName* : nom symbolique du fichier
- *PhysName* : nom physique du fichier
- *RemPhysName* : nom physique du fichier chez le partenaire
- *Label* : label du fichier
- *Origin* : origine du fichier
- *Destination* : destination du fichier

3.2.9.3 Action sur une notification (*ApiCxNetActionNot*)

Description

Cette fonction permet à l'application d'agir sur une notification mémorisée dans le fichier des notifications du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *CliName* : nom symbolique du client notifié
- *ReqNumber* : numéro de requête de transfert
- *Action* : action à effectuer sur la notification et peut prendre les valeurs 'APICX_ACTP' pour Purger ou 'APICX_ACTR' pour Router vers un autre client
- *NewClient* : nom symbolique du client vers lequel il faut router la notification

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_ACTION_NOT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10 Accès aux tables du moniteur

3.2.10.1 Liste des tables de Présentation PeSIT (*ApiCxNetListPPres*)

Description

Cette fonction permet à l'application d'obtenir la liste des tables de présentation du protocole PeSIT définies pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des tables
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_PPRES donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_PPRES est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms de tables dans la liste
- *TotDef* : nombre de noms de tables définies dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste retournée présente une suite de noms de tables de 50 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.2.10.2 Visualisation d'une table de Présentation PeSIT (ApiCxNetViewPPres)

Description

Cette fonction permet à l'application de visualiser la définition d'une table de présentation du protocole PeSIT pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PPRES donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PPRES est complétée de la manière suivante :

- *CompMode* : type de compression peut prendre les valeurs 'APICX_NOCOMP' pour Pas de compression, 'APICX_HCOMP' pour Horizontale, 'APICX_VCOMP' pour Verticale ou 'APICX_MCOMP' pour Mixte
- *ConcMode* : prend la valeur 'O' pour la mise en œuvre de la concaténation des messages du protocole dans les messages réseaux ou 'N' sinon
- *MultiMode* : prend la valeur 'O' pour la mise en œuvre de la concaténation des enregistrements du fichier les messages du protocole ou 'N' sinon
- *SegmMode* : prend la valeur 'O' pour la segmentation des enregistrements du fichier dans plusieurs messages du protocole ou 'N' sinon
- *TransMode* : prend la valeur 'O' pour la mise en œuvre de la translation ASCII / EBCDIC ou 'N' sinon
- *AtoEName* : nom de la table de translation ASCII -> EBCDIC
- *EtoAName* : nom de la table de translation EBCDIC -> ASCII

3.2.10.3 Ajout/Modification d'une table de Présentation PeSIT (*ApiCxNetUpdatePPres*)

Description

Cette fonction permet à l'application d'ajouter ou de modifier une table de présentation du protocole PeSIT pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *CompMode* : type de compression peut prendre les valeurs 'APICX_NOCOMP' pour Pas de compression, 'APICX_HCOMP' pour Horizontale, 'APICX_VCOMP' pour Verticale ou 'APICX_MCOMP' pour Mixte
- *ConcMode* : prend la valeur 'O' pour la mise en œuvre de la concaténation des messages du protocole dans les messages réseaux ou 'N' sinon
- *MultiMode* : prend la valeur 'O' pour la mise en œuvre de la concaténation des enregistrements du fichier les messages du protocole ou 'N' sinon
- *SegmMode* : prend la valeur 'O' pour la segmentation des enregistrements du fichier dans plusieurs messages du protocole ou 'N' sinon
- *TransMode* : prend la valeur 'O' pour la mise en œuvre de la translation ASCII / EBCDIC ou 'N' sinon
- *AtoEName* : nom de la table de translation ASCII -> EBCDIC
- *EtoAName* : nom de la table de translation EBCDIC -> ASCII

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PPRES donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10.4 Suppression d'une table de Présentation PeSIT (ApiCxNetDelPPres)

Description

Cette fonction permet à l'application de supprimer une table de présentation du protocole PeSIT du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_PPRES donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10.5 Liste des tables de Session PeSIT (*ApiCxNetListPSess*)

Description

Cette fonction permet à l'application d'obtenir la liste des tables de session du protocole PeSIT définies pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : l'adresse d'un emplacement mémoire pour recevoir la liste des tables
- *ListSize* : la taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_PSESS donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_PSESS est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms de tables dans la liste
- *TotDef* : nombre de noms de tables définies dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste retournée présente une suite de noms de tables de 50 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.2.10.6 Visualisation d'une table de Session PeSIT (*ApiCxNetViewPSess*)

Description

Cette fonction permet à l'application de visualiser la définition d'une table de session du protocole PeSIT pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *SessName* : le nom de la table de session sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PSESS donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PSESS est complétée de la manière suivante :

- *MsgSz* : taille maximum des messages réseau en octets
- *SyncInt* : taille de l'intervalle de synchronisation en kilo octets
- *Direction* : prend la valeur 'APICX_DIRR' pour Réception seulement, 'APICX_DIRT' pour Transmission seulement ou 'APICX_DIR' pour Réception et Transmission
- *CrcMode* : prend la valeur 'O' pour la mise en œuvre du mécanisme de contrôle de CRC ou 'N' sinon
- *ResyncCnt* : nombre maximum de re-synchronisations

3.2.10.7 Ajout/Modification d'une table de Session PeSIT (ApiCxNetUpdatePPres)

Description

Cette fonction permet à l'application d'ajouter ou de modifier une table de session du protocole PeSIT pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *SessName* : le nom de la table de session sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *MsgSz* : taille maximum des messages réseau en octets
- *SyncInt* : taille de l'intervalle de synchronisation en kilo octets
- *Direction* : prend la valeur 'APICX_DIRR' pour Réception seulement, 'APICX_DIRT' pour Transmission seulement ou 'APICX_DIR' pour Réception et Transmission
- *CrcMode* : prend la valeur 'O' pour la mise en œuvre du mécanisme de contrôle de CRC ou 'N' sinon
- *ResyncCnt* : nombre maximum de re-synchronisations

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PSESS donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10.8 Suppression d'une table de Session PeSIT (ApiCxNetDelPSess)

Description

Cette fonction permet à l'application de supprimer une table de session du protocole PeSIT du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *SessName* : le nom de la table de session sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_PSESS donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10.9 Liste des tables de Présentation ETEBAC-3 (*ApiCxNetListPEtb3*)

Description

Cette fonction permet à l'application d'obtenir la liste des tables de présentation du protocole ETEBAC-3 définies pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des tables
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_PETB3 donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_PETB3 complétée de la manière suivante :

- *NbNamesRet* : nombre de noms de tables dans la liste
- *TotDef* : nombre de noms de tables définies dans le répertoire

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste retournée présente une suite de noms de tables de 50 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.2.10.10 Visualisation d'une table de Présentation ETEBAC-3 (ApiCxNetViewPEtb3)

Description

Cette fonction permet à l'application de visualiser la définition d'une table de présentation du protocole ETEBAC-3 pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PETB3 donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PETB3 est complétée de la manière suivante :

- *TransMode* : prend la valeur 'O' pour la mise en œuvre de la translation ASCII / EBCDIC ou 'N' sinon
- *AtoEName* : nom de la table de translation ASCII -> EBCDIC
- *EtoAName* : nom de la table de translation EBCDIC -> ASCII

3.2.10.11 Ajout/Modification d'une table de Présentation ETEBAC-3 (ApiCxNetUpdatePEtb3)

Description

Cette fonction permet à l'application d'ajouter ou de modifier une table de présentation du protocole ETEBAC-3 pour le moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)
- *TransMode* : prend la valeur 'O' pour la mise en œuvre de la translation ASCII / EBCDIC ou 'N' sinon
- *AtoEName* : nom de la table de translation ASCII -> EBCDIC
- *EtoAName* : nom de la table de translation EBCDIC -> ASCII

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PETB3 donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.10.12 Suppression d'une table de Présentation ETEBAC-3 (ApiCxNetDelPEtb3)

Description

Cette fonction permet à l'application de supprimer une table de présentation du protocole ETEBAC-3 du moniteur connecté.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *PresName* : le nom de la table de présentation sur au plus 50 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_DEL_PETB3 donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.2.11 Accès aux paramètres du moniteur

3.2.11.1 Visualisation des paramètres de Démarrage du moniteur (*ApiCxNetViewPrmStr*)

Description

Cette fonction permet à l'application de visualiser les paramètres de démarrage du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXN_PRM_STR* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXN_PRM_STR* est complétée de la manière suivante :

- *RestartCnt* : nombre maximum de reprises automatiques que le moniteur peut effectuer sur un transfert et peut prendre une valeur de '00' à '99'
- *WaitInt* : nombre de secondes d'attente avant d'effectuer une reprise automatique et peut prendre une valeur de '000' à '300'
- *NetworkTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant un transfert et peut prendre une valeur de '000' à '300'
- *ClientTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant une connexion Client / Serveur et peut prendre une valeur de '000' à '300'
- *StartType* : type de démarrage du moniteur et peut prendre les valeurs 'APICX_STARTC' pour un démarrage à Chaud ou 'APICX_STARTF' pour un démarrage à Froid

3.2.11.2 Visualisation des paramètres de Service du moniteur (*ApiCxNetViewPrmSrv*)

Description

Cette fonction permet à l'application de visualiser les paramètres de service du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_SRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_SRV est complétée de la manière suivante :

- *SrvState* : prend la valeur 'O' si le moniteur est installé en temps que service de Windows NT ou 'N' sinon
- *Services* : liste des services dépendants

La liste retournée présente une suite de noms de service de 127 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.2.11.3 Visualisation des paramètres Fichier du moniteur (*ApiCxNetViewPrmFil*)

Description

Cette fonction permet à l'application de visualiser les paramètres fichier du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_FIL donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_FIL est complétée de la manière suivante :

- *InitJnl* : prend la valeur 'O' si le fichier journal doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *JnlSize* : nombre maximum d'enregistrements que le fichier journal doit contenir après réinitialisation
- *InitLog* : prend la valeur 'O' si le fichier des messages doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *LogSize* : nombre maximum d'enregistrements que le fichier des messages doit contenir après réinitialisation
- *Language* : non utilisé dans cette version. Le langage utilisé est défini en fonction des paramètres du système et l'Anglais est utilisé par défaut si ce langage n'est pas supporté par Connect:Express.
- *Trace* : prend la valeur 'O' pour l'activation de la trace interne ou 'N' sinon
- *InitNot* : prend la valeur 'O' si le fichier des notifications doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *NotSize* : nombre maximum d'enregistrements que le fichier des notifications doit contenir après réinitialisation

3.2.11.4 Visualisation des paramètres d'Autorisation du moniteur (*ApiCxNetViewPrmAut*)

Description

Cette fonction permet à l'application de visualiser les paramètres d'autorisation du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

Note :

Ces paramètres d'autorisation logicielle fournis à partir des mots clé ALIAS et NUMERO présents dans le fichier d'initialisation, ne sont plus valides que pour les anciennes clés d'autorisation et ne sont maintenus que pour des raisons de compatibilité. Les nouvelles clés d'autorisation sont fournies par Sterling Commerce sous forme d'un fichier license.key (Voir paragraphe suivant).

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_AUT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_AUT est complétée de la manière suivante :

- *Alias* : alias faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *Number* : numéro faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)

3.2.11.5 Visualisation des éléments de la clé logicielle (*Licence*) (*ApiCxNetListLicense*)

Description

Cette fonction permet à l'application d'obtenir la liste des paramètres de la clé d'autorisation logicielle validés par le moniteur à partir du fichier license.key d'autorisation fourni par Sterling Commerce et présent dans le répertoire d'installation du moniteur. L'application fournit en paramètre d'entrée l'adresse d'un buffer préalablement alloué, d'une taille suffisante pour recevoir la liste des éléments de la licence. Un buffer de 4Ko est suffisant.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir les éléments de la liste
- *ListSize* : La taille du buffer

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_LICENSE donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, le buffer ListAddr contient la liste des éléments de la clé sous la forme suivante :

line000=élément₀,line001=élément₁, ...,line00n=élément_n

3.2.11.6 Visualisation des paramètres de notification du moniteur (ApiCxNetViewPrmNot)

Description

Cette fonction permet à l'application de visualiser les paramètres de notification du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_NOT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_NOT est complétée de la manière suivante :

- *Notify* : prend la valeur 'O' si la notification globale est mise en service ou 'N' sinon
- *CliName* : nom symbolique du client à notifier sur 8 caractères en majuscules et complété par des espaces

3.2.11.7 Visualisation des paramètres TCP/IP du moniteur (*ApiCxNetViewPrmIp*)

Description

Cette fonction permet à l'application de visualiser les paramètres TCP/IP du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_IP donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_IP est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau TCP/IP est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse IP de l'appelant doit être effectué ou 'N' sinon
- *TrfPort* : numéro de port IP pour recevoir les appels entrants pour les transferts sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *CliPort* : numéro de port IP pour recevoir les appels entrants pour les connexions Clientes sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL TCP/IP présentant une interface Windows Sockest V1.1 à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.11.8 Visualisation des paramètres SNA LU6.2 du moniteur (*ApiCxNetViewPrmSna*)

Description

Cette fonction permet à l'application de visualiser les paramètres SNA LU6.2 du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_SNA donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_SNA est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau LU6.2 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle du nom de LU de l'appelant doit être effectué ou 'N' sinon
- *LuName* : nom de la LU locale pour recevoir les appels entrants sur 8 caractères en majuscules et complété par des espaces
- *AppcDllNm* : nom de la DLL APPC de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *CsvDllNm* : nom de la DLL CSV de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.11.9 Visualisation des paramètres X.25 du moniteur (ApiCxNetViewPrmX25)

Description

Cette fonction permet à l'application de visualiser les paramètres X.25 du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_PRM_X25 donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_PRM_X25 est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau X.25 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse de l'appelant doit être effectué ou 'N' sinon
- *LocAddr* : sous-adresse locale pour recevoir les appels entrants sur au plus 15 caractères et terminé par au moins un caractère nul (\0)
- *PortNb* : numéro de port pour recevoir les appels entrants sur au plus 2 caractères et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL EX25 EICON à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.11.10 Visualisation des paramètres Canal Nommé du moniteur (*ApiCxNetViewPrmNp*)

Description

Cette fonction permet à l'application de visualiser les paramètres canal nommé du moniteur connecté. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXN_PRM_NP* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXN_PRM_NP* est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau Canal Nommé (Named Pipe) est mis en service ou 'N' sinon
- *NpName* : nom complet du canal nommé pour les connexions Clientes sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.2.11.11 Liste des serveurs SSL (*ApiCxNetListSslparmSrv*)

Description

Cette fonction permet à l'application d'obtenir la liste des paramètres de serveurs SSL définis pour le moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *ListAddr* : L'adresse d'un emplacement mémoire pour recevoir la liste des serveurs SSL
- *ListSize* : La taille de la liste

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_SSLLPARMSRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_LIST_SSLLPARMSRV est complétée de la manière suivante :

- *NbNamesRet* : nombre de noms symboliques de serveurs SSL dans la liste
- *TotDef* : nombre de noms symboliques de serveurs SSL définis

Si le nombre de noms définis est supérieur au nombre de noms dans la liste, cela indique que la taille de l'emplacement mémoire donné en entrée n'est pas suffisant pour mémoriser toute la liste.

La liste obtenue est composée de noms symboliques de serveurs SSL sur 8 caractères complétés avec des espaces si nécessaire.

3.2.11.12 Visualisation des paramètres d'un serveur SSL (*ApiCxNetViewSslparmSrv*)

Description

Cette fonction permet à l'application de visualiser la définition des paramètres d'un serveur SSL pour le moniteur connecté. Elle n'est applicable que si l'option SSL est activée pour le moniteur. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec un moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise
- *Name* : le nom symbolique du serveur SSL sur 8 caractères en majuscules, complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_SSLLPARMSRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXN_SSLLPARMSRV est complétée de la manière suivante :

- *Enabled* : état du serveur ('1' : En service, '0' : Hors service)
- *StoreProvider* : Fournisseur de certificat. Chaîne de caractères 'STORE_PROV_SYSTEM' terminée par un zéro binaire.
- *StoreLocation* : Emplacement du certificat. Chaîne de caractères terminée par un zéro binaire pouvant prendre l'une des valeurs : 'SYSTEM_STORE_LOCAL_MACHINE', 'SYSTEM_STORE_SERVICES' ou 'SYSTEM_STORE_CURRENT_USER'.
- *StoreName* : Chaîne de caractères 'My' terminée par un zéro binaire .
- *SubjectDn* : Distinguished name objet du certificat du serveur. Chaîne de caractères terminée par un zéro binaire.
- *IssuerDn* : Distinguished name émetteur du certificat du serveur. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *Protocol* : Version de protocole SSL utilisée ('1' :TLSV1,'3' :SSLV3, '2' :SSLV2)
- *CipherSuites* : Suites de chiffrement. Chaîne éventuellement vide formée d'une liste de n° de 2 digits chacun, indiquant les suites de chiffrement à utiliser. La chaîne ainsi formée est terminée par un zéro binaire. Les numéros suivant peuvent être utilisés :
'00' TLS_RSA_WITH_RC4_128_MD5
'01' TLS_RSA_WITH_RC4_128_SHA
'02' TLS_RSA_WITH_3DES_EDE_CBC_SHA,
'03' TLS_RSA_WITH_DES_CBC_SHA,
'06' SSL CK_RC4_128_WITH_MD5,
'07' SSL CK_DES_64_CBC_WITH_MD5,
'08' SSL CK_RC2_128_CBC_WITH_MD5
Exemple : 000103\0 indique les suites 00, 01 et 03.
- *Trace* : Indicateur de trace ('0' : Aucune, '1' : Trace partielle, '2' : Trace complète)
- *ServerPort* : Port TCP/IP du serveur (Maximum 5 digits terminés par un zéro binaire)

3.2.11.13 Visualisation des paramètres d'un serveur SSL (*ApiCxNetViewSslparmSrvEx*)

Description

Cette fonction est semblable à la précédente, mais la structure contient en plus un champ :

- *SslBytesHeader* : Indique que les données non encryptées sont préfixées avec 2 octets de longueur ('0' : Pas d'octets de longueur, '1' : Il y a 2 octets de longueur)

La structure est nommée APICXN_SSLPARMSRV_EX.

3.3 Les fonctions d'accès direct

3.3.1 Paramètres de Démarrage du moniteur

3.3.1.1 Visualisation des paramètres de Démarrage du moniteur (*ApiCxShrViewPrmStr*)

Description

Cette fonction permet à l'application de visualiser les paramètres de démarrage du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_STR donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows

Si la fonction a été exécutée correctement, la structure APICXS_PRM_STR est complétée de la manière suivante :

- *RestartCnt* : nombre maximum de reprises automatiques que le moniteur peut effectuer sur un transfert et peut prendre une valeur de '00' à '99'
- *WaitInt* : nombre de secondes d'attente avant d'effectuer une reprise automatique et peut prendre une valeur de '000' à '300'
- *NetworkTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant un transfert et peut prendre une valeur de '000' à '300'
- *ClientTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant une connexion Client / Serveur et peut prendre une valeur de '000' à '300'
- *StartType* : type de démarrage du moniteur et peut prendre les valeurs 'APICX_STARTC' pour un démarrage à Chaud ou 'APICX_STARTF' pour un démarrage à Froid

3.3.1.2 Modification des paramètres de Démarrage du moniteur (*ApiCxShrUpdatePrmStr*)

Description

Cette fonction permet à l'application de modifier les paramètres de démarrage du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *RestartCnt* : nombre maximum de reprises automatiques que le moniteur peut effectuer sur un transfert et peut prendre une valeur de '00' à '99'
- *WaitInt* : nombre de secondes d'attente avant d'effectuer une reprise automatique et peut prendre une valeur de '000' à '300'
- *NetworkTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant un transfert et peut prendre une valeur de '000' à '300'
- *ClientTimer* : nombre de secondes d'échéance de temporisation sur le réseau pendant une connexion Client / Serveur et peut prendre une valeur de '000' à '300'
- *StartType* : type de démarrage du moniteur et peut prendre les valeurs 'APICX_STARTC' pour un démarrage à Chaud ou 'APICX_STARTF' pour un démarrage à Froid

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_STR donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows

3.3.2 Paramètres de Service du moniteur

3.3.2.1 Visualisation des paramètres de Service du moniteur (*ApiCxShrViewPrmSrv*)

Description

Cette fonction permet à l'application de visualiser les paramètres de service du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_SRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXS_PRM_SRV est complétée de la manière suivante :

- *SrvState* : prend la valeur 'O' si le moniteur est installé en temps que service de Windows NT ou 'N' sinon
- *Services* : liste des services dépendants

La liste retournée présente une suite de noms de service de 127 caractères maximum, séparés par un caractère nul (\0) et dont le dernier est suivi par au moins deux caractères nuls (\0\0).

3.3.2.2 Modification des paramètres de Service du moniteur

La modification des paramètres du Service C :X ne peut s'effectuer qu'au travers de l'utilitaire d'installation/Désinstallation du Service C :X.

Cet utilitaire est documenté dans le '*Guide de l'installation et des Utilitaires de Connect:Express Windows*'.

3.3.3 Paramètres Fichier du moniteur

3.3.3.1 Visualisation des paramètres Fichier du moniteur (*ApiCxShrViewPrmFil*)

Description

Cette fonction permet à l'application de visualiser les paramètres fichier du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_FIL* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXS_PRM_FIL* est complétée de la manière suivante :

- *InitJnl* : prend la valeur 'O' si le fichier journal doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *JnlSize* : nombre maximum d'enregistrements que le fichier journal doit contenir après réinitialisation
- *InitLog* : prend la valeur 'O' si le fichier des messages doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *LogSize* : nombre maximum d'enregistrements que le fichier des messages doit contenir après réinitialisation
- *Language* : non utilisé dans cette version. Le langage utilisé est défini en fonction des paramètres du système et l'Anglais est utilisé par défaut si ce langage n'est pas supporté par Connect:Express.
- *Trace* : prend la valeur 'O' pour l'activation de la trace interne ou 'N' sinon
- *InitNot* : prend la valeur 'O' si le fichier des notifications doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *NotSize* : nombre maximum d'enregistrements que le fichier des notifications doit contenir après réinitialisation

3.3.3.2 Modification des paramètres Fichier du moniteur (*ApiCxShrUpdatePrmFil*)

Description

Cette fonction permet à l'application de modifier les paramètres fichier du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *InitJnl* : prend la valeur 'O' si le fichier journal doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *JnlSize* : nombre maximum d'enregistrements que le fichier journal doit contenir après réinitialisation
- *InitLog* : prend la valeur 'O' si le fichier des messages doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *LogSize* : nombre maximum d'enregistrements que le fichier des messages doit contenir après réinitialisation
- *Language* : non utilisé dans cette version. Le langage utilisé est défini en fonction des paramètres du système et l'Anglais est utilisé par défaut si ce langage n'est pas supporté par Connect:Express.
- *Trace* : prend la valeur 'O' pour l'activation de la trace interne ou 'N' sinon
- *InitNot* : prend la valeur 'O' si le fichier des notifications doit être réinitialisé lors du lancement du moniteur ou 'N' sinon
- *NotSize* : nombre maximum d'enregistrements que le fichier des notifications doit contenir après réinitialisation

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_FIL donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.4 Paramètres d'Autorisation du moniteur

3.3.4.1 Visualisation des paramètres d'Autorisation du moniteur (*ApiCxShrViewPrmAut*)

Description

Cette fonction permet à l'application de visualiser les paramètres d'autorisation du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_AUT* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXS_PRM_AUT* est complétée de la manière suivante :

- *Alias* : alias faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *Number* : numéro faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)

3.3.4.2 Modification des paramètres d'Autorisation du moniteur (*ApiCxShrUpdatePrmAut*)

Description

Cette fonction permet à l'application de modifier les paramètres d'autorisation du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

Ces paramètres d'autorisation logicielle fournis à partir des mots clé ALIAS et NUMERO présents dans le fichier d'initialisation, ne sont plus valides que pour les anciennes clés d'autorisation et ne sont maintenus que pour des raisons de compatibilité. Les nouvelles clés d'autorisation sont fournies par Sterling Commerce sous forme d'un fichier license.key.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *Alias* : alias faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)
- *Number* : numéro faisant partie de la clé d'autorisation du produit Connect:Express sur au plus 80 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_AUT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.4.3 Chargement du fichier license.key par le moniteur (*ApiCxNetLoadLicense*)

Description

Cette fonction permet à l'application de demander au moniteur de charger les paramètres contenus dans un fichier d'autorisation license.key, préalablement placé dans le répertoire d'installation du moniteur. Ce fichier contient la clé de protection logicielle et indique au moniteur les différentes options autorisées ainsi que leur date d'expiration. Il est fourni par Sterling Commerce. Dans le cas où une clé d'autorisation arrive à expiration ou si une nouvelle composante a été souscrite, il est possible de mettre à jour la clé logicielle sans avoir à arrêter le moniteur en utilisant cette fonction.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès ainsi que la fonction d'établissement de connexion avec le moniteur avec succès.

En entrée

L'application indique :

- *CxId* : l'identifiant de la connexion Client/Serveur avec le moniteur de transfert
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LOAD_LICENSE donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.5 Paramètres de Notification du moniteur

3.3.5.1 Visualisation des paramètres de Notification du moniteur (*ApiCxShrViewPrmNot*)

Description

Cette fonction permet à l'application de visualiser les paramètres de notification du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_NOT* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXS_PRM_NOT* est complétée de la manière suivante :

- *Notify* : prend la valeur 'O' si la notification globale est mise en service ou 'N' sinon
- *CliName* : nom symbolique du client à notifier sur 8 caractères en majuscules et complété par des espaces

3.3.5.2 Modification des paramètres de Notification du moniteur (*ApiCxShrUpdatePrmNot*)

Description

Cette fonction permet à l'application de modifier les paramètres de notification du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *Notify* : prend la valeur 'O' si la notification globale est mise en service ou 'N' sinon
- *CliName* : nom symbolique du client à notifier sur 8 caractères en majuscules et complété par des espaces

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_NOT donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.6 Paramètres TCP/IP du moniteur

3.3.6.1 Visualisation des paramètres TCP/IP du moniteur (*ApiCxShrViewPrmIp*)

Description

Cette fonction permet à l'application de visualiser les paramètres TCP/IP du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_IP donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXS_PRM_IP est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau TCP/IP est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse IP de l'appelant doit être effectué ou 'N' sinon
- *TrfPort* : numéro de port IP pour recevoir les appels entrants pour les transferts sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *CliPort* : numéro de port IP pour recevoir les appels entrants pour les connexions Clientes sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL TCP/IP présentant une interface Windows Sockest V1.1 à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.3.6.2 Modification des paramètres TCP/IP du moniteur (*ApiCxShrUpdatePrmIp*)

Description

Cette fonction permet à l'application de modifier les paramètres TCP/IP du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *State* : prend la valeur 'O' si le réseau TCP/IP est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse IP de l'appelant doit être effectué ou 'N' sinon
- *TrfPort* : numéro de port IP pour recevoir les appels entrants pour les transferts sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *CliPort* : numéro de port IP pour recevoir les appels entrants pour les connexions Clientes sur au plus 5 caractères numériques et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL TCP/IP présentant une interface Windows Sockest V1.1 à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_IP donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.7 Paramètres SNA/LU6.2 du moniteur

3.3.7.1 Visualisation des paramètres SNA/LU6.2 du moniteur (*ApiCxShrViewPrmSna*)

Description

Cette fonction permet à l'application de visualiser les paramètres SNA du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_SNA donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXS_PRM_SNA est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau LU6.2 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle du nom de LU de l'appelant doit être effectué ou 'N' sinon
- *LuName* : nom de la LU locale pour recevoir les appels entrants sur 8 caractères en majuscules et complété par des espaces
- *AppcDllNm* : nom de la DLL APPC de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *CsvDllNm* : nom de la DLL CSV de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.3.7.2 Modification des paramètres SNA/LU6.2 du moniteur (*ApiCxShrUpdatePrmSna*)

Description

Cette fonction permet à l'application de modifier les paramètres SNA du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *State* : prend la valeur 'O' si le réseau LU6.2 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle du nom de LU de l'appelant doit être effectué ou 'N' sinon
- *LuName* : nom de la LU locale pour recevoir les appels entrants sur 8 caractères en majuscules et complété par des espaces
- *AppcDllNm* : nom de la DLL APPC de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)
- *CsvDllNm* : nom de la DLL CSV de SNA SERVER à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_SNA donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.8 Paramètres X.25 du moniteur

3.3.8.1 Visualisation des paramètres X.25 du moniteur (*ApiCxShrViewPrmX25*)

Description

Cette fonction permet à l'application de visualiser les paramètres X.25 du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_X25* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXS_PRM_X25* est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau X.25 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse de l'appelant doit être effectué ou 'N' sinon
- *LocAddr* : sous-adresse locale pour recevoir les appels entrants sur au plus 15 caractères et terminé par au moins un caractère nul (\0)
- *PortNb* : numéro de port pour recevoir les appels entrants sur au plus 2 caractères et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL EX25 EICON à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.3.8.2 Modification des paramètres X.25 du moniteur (*ApiCxShrUpdatePrmX25*)

Description

Cette fonction permet à l'application de modifier les paramètres X.25 du fichier d'initialisation indiqué. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *State* : prend la valeur 'O' si le réseau X.25 est mis en service ou 'N' sinon
- *CheckAddr* : prend la valeur 'O' si le contrôle de l'adresse de l'appelant doit être effectué ou 'N' sinon
- *LocAddr* : sous-adresse locale pour recevoir les appels entrants sur au plus 15 caractères et terminé par au moins un caractère nul (\0)
- *PortNb* : numéro de port pour recevoir les appels entrants sur au plus 2 caractères et terminé par au moins un caractère nul (\0)
- *DllName* : nom de la DLL EX25 EICON à utiliser sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_X25* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.9 Paramètres Canal Nommé du moniteur

3.3.9.1 Visualisation des paramètres Canal Nommé du moniteur (*ApiCxShrViewPrmNp*)

Description

Cette fonction permet à l'application de visualiser les paramètres Canal Nommé du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure *APICXS_PRM_NP* donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure *APICXS_PRM_NP* est complétée de la manière suivante :

- *State* : prend la valeur 'O' si le réseau Canal Nommé (Named Pipe) est mis en service ou 'N' sinon
- *NpName* : nom complet du canal nommé pour les connexions Clientes sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

3.3.9.2 Modification des paramètres Canal Nommé du moniteur (*ApiCxShrUpdatePrmNp*)

Description

Cette fonction permet à l'application de modifier les paramètres Canal Nommé du fichier d'initialisation indiqué.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *State* : prend la valeur 'O' si le réseau Canal Nommé (Named Pipe) est mis en service ou 'N' sinon
- *NpName* : nom complet du canal nommé pour les connexions Clientes sur au plus 127 caractères et terminé par au moins un caractère nul (\0)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_NP donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.10 Paramètres des serveurs SSL

3.3.10.1 Liste des serveurs SSL (*ApiCxShrListSslparmSrv*)

Description

Cette fonction permet à l'application d'obtenir la liste des serveurs SSL définis dans le fichier d'initialisation indiqué. Elle n'est applicable que si l'option SSL est activée pour le moniteur. Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXN_LIST_SSLPARMSRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXS_LIST_SSLPARMSRV est complétée de la manière suivante :

- *ListBuffer* : Buffer contenant la liste des noms symboliques des serveurs SSL
- *NbNamesRet* : nombre de noms symboliques de serveurs SSL dans la liste
- *TotDef* : nombre de noms symboliques de serveurs SSL définis

La liste obtenue est composée de noms symboliques de serveurs SSL sur 8 caractères complétés avec des espaces si nécessaire.

3.3.10.2 Visualisation des paramètres d'un serveur SSL (*ApiCxShrViewSslparmSrv*)

Description

Cette fonction permet à l'application de visualiser les paramètres d'un serveur SSL du fichier d'initialisation indiqué. Elle n'est applicable que si l'option SSL est activée pour le moniteur.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *Name* : nom symbolique du serveur SSL (au maximum 8 caractères majuscules complétés à blanc)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_PRM_NP donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

Si la fonction a été exécutée correctement, la structure APICXS_SSLLPARMSRV est complétée de la manière suivante :

- *Enabled* : état du serveur ('1' : En service, '0' : Hors service)
- *StoreProvider* : Fournisseur de certificat. Chaîne de caractères 'STORE_PROV_SYSTEM' terminée par un zéro binaire.
- *StoreLocation* : Emplacement du certificat. Chaîne de caractères terminée par un zéro binaire pouvant prendre l'une des valeurs : 'SYSTEM_STORE_LOCAL_MACHINE', 'SYSTEM_STORE_SERVICES' ou 'SYSTEM_STORE_CURRENT_USER'.
- *StoreName* : Chaîne de caractères 'My' terminée par un zéro binaire .
- *SubjectDn* : Distinguished name objet du certificat du serveur. Chaîne de caractères terminée par un zéro binaire.
- *IssuerDn* : Distinguished name émetteur du certificat du serveur. Chaîne de caractères terminée par un zéro binaire. Champ facultatif.
- *Protocol* : Version de protocole SSL utilisée ('1' :TLSV1,'3' :SSLV3, '2' :SSLV2)
- *CipherSuites* : Suites de chiffrement. Chaîne éventuellement vide formée d'une liste de n° de 2 digits chacun, indiquant les suites de chiffrement à utiliser. La chaîne ainsi formée est terminée par un zéro binaire. Les numéros suivant peuvent être utilisés :
'00' TLS_RSA_WITH_RC4_128_MD5
'01' TLS_RSA_WITH_RC4_128_SHA
'02' TLS_RSA_WITH_3DES_EDE_CBC_SHA,
'03' TLS_RSA_WITH_DES_CBC_SHA,
'06' SSL CK_RC4_128_WITH_MD5,
'07' SSL CK_DES_64_CBC_WITH_MD5,
'08' SSL CK_RC2_128_CBC_WITH_MD5
Exemple : 000103\0 indique les suites 00, 01 et 03.
- *Trace* : Indicateur de trace ('0' : Aucune, '1' : Trace partielle, '2' : Trace complète)
- *ServerPort* : Port TCP/IP du serveur (Maximum 5 digits terminés par un zéro binaire)

3.3.10.3 Visualisation des paramètres d'un serveur SSL (*ApiCxShrViewSslparmSrvEx*)

Description

Cette fonction est semblable à la précédente, mais contient en plus un champ :

- *SslBytesHeader* : Indique que les données non encryptées sont préfixées avec 2 octets de longueur ('0' : Pas d'octets de longueur, '1' : Il y a 2 octets de longueur)

La structure est nommée APICXS_SSLPARAMSRV_EX.

3.3.10.4 Modification des paramètres d'un serveur SSL (*ApiCxShrUpdateSslparmSrv*)

Description

Cette fonction permet à l'application de modifier les paramètres d'un serveur SSL du fichier d'initialisation indiqué. Elle n'est applicable que si l'option SSL est activée pour le moniteur.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *Name* : le nom symbolique du serveur SSL sur 8 caractères en majuscules, complété par des espaces. Si le serveur SSL n'existe pas au répertoire c'est un ajout et, s'il existe déjà, c'est une modification.
- *Enabled* : état du serveur ('1' : En service, '0' : Hors service)
- *StoreProvider* : Fournisseur de certificat. Chaîne de caractères 'STORE_PROV_SYSTEM' terminée par un zéro binaire.
- *StoreLocation* : Emplacement du certificat. Chaîne de caractères terminée par un zéro binaire pouvant prendre l'une des valeurs : 'SYSTEM_STORE_LOCAL_MACHINE', 'SYSTEM_STORE_SERVICES' ou 'SYSTEM_STORE_CURRENT_USER'.
- *StoreName* : Chaîne de caractères 'My' terminée par un zéro.
- *SubjectDn* : Distinguished name objet du certificat du serveur. Chaîne de caractères terminée par un zéro binaire.
- *IssuerDn* : Distinguished name émetteur du certificat du serveur. Chaîne de caractères terminée par un zéro binaire.
- *Protocol* : Version de protocole SSL utilisée ('1' : TLSV1, '3' : SSLV3, '2' : SSLV2)
- *CipherSuites* : Suites de chiffrement. Renseigner une chaîne éventuellement vide formée d'une liste de n° de 2 digits chacun, indiquant les suites de chiffrement à utiliser. La chaîne ainsi formée doit être terminée par un zéro binaire. Les numéros suivant peuvent être utilisés :
 - '00' TLS_RSA_WITH_RC4_128_MD5
 - '01' TLS_RSA_WITH_RC4_128_SHA
 - '02' TLS_RSA_WITH_3DES_EDE_CBC_SHA,
 - '03' TLS_RSA_WITH_DES_CBC_SHA,
 - '06' SSL CK RC4_128_WITH_MD5,
 - '07' SSL CK DES_64_CBC_WITH_MD5,
 - '08' SSL CK RC2_128_CBC_WITH_MD5
 Exemple : 000103\0 indique les suites 00, 01 et 03.
- *Trace* : Indicateur de trace ('0' : Aucune, '1' : Trace partielle, '2' : Trace complète)
- *ServerPort* : Port TCP/IP du serveur (Maximum 5 digits terminés par un zéro binaire)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_SSLPARAMSRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).

3.3.10.5 Modification des paramètres d'un serveur SSL (*ApiCxShrUpdateSslparmSrvEx*)

Description

Cette fonction est semblable à la précédente, mais contient en plus un champ :

- *SslBytesHeader* : Indique que les données non encryptées sont préfixées avec 2 octets de longueur ('0' : Pas d'octets de longueur, '1' : Il y a 2 octets de longueur)

La structure est nommée APICXS_SSLPARAMSRV_EX.

3.3.10.6 Suppression d'un serveur SSL (*ApiCxShrDelSslparmSrv*)

Description

Cette fonction permet à l'application de supprimer la définition d'un serveur SSL du fichier d'initialisation indiqué. Elle n'est applicable que si l'option SSL est activée pour le moniteur.

Pour appeler cette fonction, l'application doit avoir exécuté au préalable la fonction d'initialisation de l'API avec succès.

En entrée

L'application indique :

- *CliName* : nom symbolique du client
- *CliPsw* : mot de passe du client
- *IniFileName* : nom du fichier d'initialisation du moniteur
- *Version* : la version de l'API qu'elle utilise
- *Name* : nom symbolique du serveur SSL (au maximum 8 caractères majuscules complétés à blanc)

En sortie

Le code retour direct de la fonction indique:

-1 : l'adresse de la structure APICXS_DEL_SSLPARAMSRV donnée par l'application est incorrecte

0 : la fonction a échoué, voir le code retour API

1 : la fonction a été exécutée correctement

Les champs *ApiRc*, *SysRc*, *TcpRc* et *TomRc* sont documentés dans le Guide utilisateur de Connect:Express Windows. Le code *TomRc* est accompagné du libellé de l'erreur donné par le moniteur (*LibErr*).