

UTILISATION DE LA BIBLIOTHEQUE DE FONCTIONS DEKOLIB.DLL SOUS DELPHI

Table des matières

UTILISATION DE LA BIBLIOTHEQUE DE FONCTIONS DEKOLIB.DLL SOUS DELPHI.....	1
I. INTRODUCTION.....	1
II. PRELIMINAIRES.....	2
III. PRESENTATION DES FONCTIONS.....	2
1. Fonction de création de l'objet DekoAutomation.....	2
2. Propriété PipeName : initialisation du Nom du Canal Nommé.....	3
3. Fonction de connexion	3
4. Envoyer une commande vers le Deko.....	3
5. Fonction de déconnexion du Deko.....	4
6. Autres Fonctions.....	4
IV. CONCLUSION.....	4

I. INTRODUCTION

Au cours de ce tutoriel vous allez apprendre à utiliser la bibliothèque de fonction DekoLib.dll qui permet d'envoyer des requêtes vers un ordinateur équipée d'une Application Deko d'Avid. Vous

verrez que son utilisation est des plus simples et en plus elle permet de s'affranchir de l'utilisation de la technologie des canaux nommés (« named pipes ») assez complexe à mettre en oeuvre. L'application Deko est utilisée dans le secteur de l'audiovisuel pour automatiser la diffusion de graphiques en temps réel. Je l'ai utilisé dans le cadre d'une application permettant aux journalistes de visualiser des tableaux de résultats (Type Tableau Classement Ligue 1) depuis leur poste client, afin de les vérifier avant leur passage à l'antenne. Ce tutoriel est avant tout une initiation à l'utilisation de cette bibliothèque. L'étendue des fonctions qu'elle propose est beaucoup plus conséquente. Mais cette introduction vous permettra d'aller plus loin ensuite...

II. PRELIMINAIRES

Ce tutoriel a pour but de vous initier dans l'utilisation de la dll DekoLib.dll. Pour pouvoir l'utiliser il faut cependant avoir installé l'application Deko sur votre ordinateur. Ce fichier est ensuite disponible dans le répertoire bin de votre dossier d'installation. Une fois que vous avez localisé ce fichier il vous reste à accomplir les tâches suivantes :

- Création d'un nouveau projet Win32 que vous nommerez selon votre convenance;
- Importation de la bibliothèque de type Deko Automation Interface lib (qui doit apparaître dans la fenêtre des bibliothèques de types recensées lorsque vous avez sélectionné Composant -> Importer un composant.. et ensuite sélectionné l'option Importer une bibliothèque de type);

Une fois cette opération réalisée Delphi créer un fichier DekoLib99_TLB.pas dans votre projet. Vous êtes alors prêt à utiliser les fonctions incluses dans cette bibliothèque.

Une chose à faire cependant est de créer une nouvelle fiche que vous nommerez comme il vous plaira. Bien entendu dans votre fiche il faut rajouter la référence à cette nouvelle unité .pas en utilisant le mot-clé *uses* :

uses dekoLib99_TLB;

Voilà maintenant nous sommes prêt. Cependant nous allons faire un petit tour d'horizon sur les fonctions qui composent cette bibliothèque de fonction.

III. PRESENTATION DES FONCTIONS

La philosophie de cette bibliothèque est de créer un objet DekoAutomation qui va nous permettre de nous connecter, de nous déconnecter mais aussi d'envoyer des requêtes d'automation vers le serveur Deko. Voici les principales fonctions et propriétés qui nous permettent de dialoguer avec un serveur.

1. Fonction de création de l'objet DekoAutomation

C'est la fonction **Create** qui permet de créer un objet DekoAutomation en utilisant l'interface CoDekoAutomation comme ceci :

DekoAutomation1 := CoDekoAutomation.Create;

DekoAutomation1 étant de type DekoAutomation.

2. Propriété PipeName : initialisation du Nom du Canal Nommé

Comme je vous le disais plus haut la communication avec un Serveur Deko utilise la technologie des canaux nommés. Il est donc impératif d'initialiser cette propriété avant de se connecter et d'envoyer des requêtes :

DekoAutomation1.PipeName := '\\ ' + addIP + '\pipe\' + nomPipe;

L'adresse IP est celle de votre serveur Deko.

Le nom du Pipe est celui que vous avez créé sur votre serveur Deko.

3. Fonction de connexion

Une fois votre objet créé et votre propriété **PipeName** initialisée, vous allez pouvoir envoyer des ordres à votre serveur Deko. Mais pour ce faire il va falloir tout d'abord établir une connexion avec ce dernier. C'est la fonction **ConnectToDeko** qui réalise cette opération :

res := DekoAutomation1.ConnectToDeko;

En fonction du résultat retourné par cette fonction il est possible de savoir si cette opération a bien aboutie :

if res <> -1 then

Statusbar1.Panels[0].Text := 'Communication à travers le canal en cours...';

4. Envoyer une commande vers le Deko

Pour envoyer une commande vers le serveur Deko il suffit d'initialiser la propriété **Command** de l'objet **DekoAutomation1**, comme ceci:

DekoAutomation1.Command := Chaine;

avec Chaine qui est votre requête de type Deko. Par exemple :

Chaine := 'M\c/typedemo\\' + #10#13;

Il est important de terminer votre chaîne de caractères par l'emploi de #10#13 ce qui permet au serveur de repérer la fin de votre requête...

Le fait d'initialiser la propriété **Command** ne suffit pas pour envoyer votre requête vers le serveur. Il faut l'envoyer et ceci se fait par l'intermédiaire de la fonction **SendToDeko**, de la façon suivante :

DekoAutomation1.SendToDeko;

Bien sûr cette opération devra être répétée pour chaque envoi de commande.

5. Fonction de déconnexion du Deko

La déconnexion s'effectue à partir de la fonction *DisconnectFromDeko* :

DekoAutomation1.DisconnectFromDeko;

Cette opération est nécessaire pour libérer l'accès au pipe pour un autre appel issu d'un autre client...

6. Autres Fonctions

Je vous ai introduit les principales fonctions dont vous avez besoin pour communiquer avec votre machine serveur Deko. Mais cette bibliothèque est beaucoup plus riche. En effet vous avez à votre disposition des fonctions permettant de gérer des fichiers macro comme *Read_File*. Mais aussi vous pourrez gérer des effets avec les fonctions *Effect_Setup*, *Effect_Build*, ainsi que *Effect_Trigger*.

IV. CONCLUSION

Vous avez pu voir comment utiliser la bibliothèque de fonctions fournie avec votre application Deko. Vous avez pu voir que la philosophie (technologie objet) utilisée par celle-ci rend la communication avec un serveur Deko beaucoup plus simple à réaliser, que s'il fallait passer par la technologie des pipes nommées. Ceci n'est qu'une introduction, mais comme je le disais cette bibliothèque permet de concevoir des applications clientes très intéressantes pour vérifier des informations critiques avant leur diffusion.