



**Spécifications pour l'utilisation de la
saisie vocale avec FlexBrowser.**

L'ActiveX voiXtreme.

wfb_voice_fr.doc

20080226 Version 1.0 FR	
-------------------------	--

Introduction

Résumé.

Ce document présente le mode d'utilisation du système de reconnaissance vocale (*ASR – Automatic Speech Recognition*) et de la synthèse vocale (*TTS – Text to Speech*) dans une application web qui sera lancée à travers *FlexBrowser*.

Ce texte, dirigé au programmeur d'applications web, présente en même temps : une introduction, la syntaxe pour l'ensemble de fonctions et des exemples.

Présentation du système

On appellera “*FlexBrowser Voice*” ou simplement “*Voice*” à l'ensemble de deux systèmes puissants qui sont en rapport avec la parole et les systèmes électroniques - *ASR (Automatic Speech Recognition)* et *TTS (Text to Speech)*- et le navigateur *FlexBrowser*.

En lignes générales, *ASR* écoute une allocution déterminée, il l'analyse selon une grammaire prédéterminée et retourne une chaîne de texte -appartenant à cette grammaire- la plus proche possible à l'allocution prononcée.

A l'inverse, le *TTS* permet de convertir en sons des ordres écrits, de sorte qu'il est possible d'écouter un texte qui se trouve sous la forme de chaîne de caractères.

Dans les deux cas tout le traitement se fait dans le terminal, sans avoir besoin d'aucune connexion avec le serveur.

FlexBrowser utilise les deux systèmes à travers un contrôle ActiveX propre qui s'appelle `voixtreme`, dont les caractéristiques sont décrites ici. Pour pouvoir utiliser le *Voice*, une page HTML (avec javascript) devrait suivre les spécifications présentées là-dessous.

voixtreme

`voixtreme.ocx` c'est le nom de l'ActiveX qui s'installe de manière indépendante de *FlexBrowser* et qui rend possible l'utilisation du *Voice*. Il y a une version différente pour chaque langage et dans chaque cas il faut installer l'ocx avec l'ensemble de données et bibliothèques nécessaires pour son fonctionnement.

A cause de la taille des composants de `voixtreme` et que les systèmes Windows Mobile 2005 ont leur système de fichiers en mémoire persistante, le *Voice* s'installe comme résistante au warmboot et coldboot, mais pas au cleanboot.

Fonctions définies, paramètres, valeurs retournées et syntaxe

L'objet à instancier a le CLSID `5FF0CA2F-A640-4F62-8EDF-C97F30B562FB`, et compte avec les méthodes suivantes :

```
int StgVoiceInit( ) ;
```

Démarre la librairie du *Voice* (ASR+TTS). Il est nécessaire d'appeler à cette fonction avant d'appeler à n'importe quelle autre fonction. Une fois démarrée, elle reste active pour toutes les pages, il n'est pas nécessaire de la redémarrer.

```
int StgVoiceDeInit( ) ;
```

Complément de `StgVoiceInit`. Il faut appeler à cette fonction quand on finit d'utiliser le *Voice*.

```
int StgVoiceSay( strToSay ) ;
```

Lit la chaîne de caractères passée en paramètre et retourne un code d'erreur. 0 (zéro) si ok.

```
int StgVoiceListen( ) ;
```

Lance le reconnaiseur de la voix de l'utilisateur. La fonction est asynchrone, c'est-à-dire que le système retourne immédiatement le contrôle et permet de suivre avec l'exécution des lignes de code suivantes. L'utilisateur est obligé à attendre la fin de la reconnaissance en évaluant la fonction `StgVoiceIsListening`, jusqu'à ce qu'elle retourne `false`. A ce moment, la chaîne résultante peut être obtenue à travers la fonction `StgVoiceGetResultString`.

```
str StgVoiceListenSync( ) ;
```

Retourne une chaîne de caractères, produit de la reconnaissance de la voix de l'utilisateur. La fonction est synchrone, c'est-à-dire que le système doit attendre le résultat de cette fonction pour continuer avec l'exécution des lignes de code suivantes.

```
int StgVoiceActivateRule( strRule ) ;
```

Active une règle de la grammaire. Cela *permet* que la fonction `Listen` retourne une chaîne de caractères générée avec cette règle.

```
int StgVoiceDeactivateRule( strRule ) ;
```

Désactive une règle de la grammaire. Cela *empêche* que la fonction `Listen` puisse retourner une chaîne de caractères générée avec cette règle.

```
int StgVoiceSetDebuggingMode(ULONG p_bShowMessageBoxes) ;
```

Permet l'activation du mode de debugging. Dans ce mode, chaque erreur d'exécution lance un message de texte communiquant l'erreur. Il est désactivé par défaut.

```
int StgVoiceIsSpeaking( ) ;
```

Retourne une valeur booléenne indiquant si le TTS est actif et en train de prononcer un texte.

```
str StgVoiceGetResultString( ) ;
```

Retourne la dernière chaîne de texte reconnue par l'ASR.

```
int StgVoiceIsListening( ) ;  
Retourne une valeur booléenne indiquant si le ASR est actif et en  
train de recevoir des inputs du système d'audio.
```

```
int StgVoiceSpell( strToSpell ) ;  
Épelle la chaîne de caractères numérique passée en paramètre et  
retourne un code d'erreur. 0 (zéro) si ok.
```

```
int StgVoiceStopListening( ) ;  
Termine abruptement le reconnaiseur de la parole sans retourner un  
résultat.
```

```
int StgVoiceStopSpeaking( ) ;  
Termine abruptement la prononciation d'une phrase.
```

Les paramètres du système vocal sont préconfigurés avec des options par défaut. Les méthodes et propriétés qui permettent une configuration plus détaillée seront ajoutées dans des versions ultérieures.

Un exemple d'utilisation

Dans l'exemple suivant on va instancier un objet comme `VoiceObj`, à travers lequel on accédera au système vocal.

Fichier `voixtremel.htm`

```
<HTML>
<HEAD>
<TITLE>Test page for Voice Object voiXtreme in FlexTools</TITLE>
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<META NAME="MobileOptimized" content="240">
<SCRIPT LANGUAGE="javascript">
    function Init()
    {
        VoiceObj.StgVoicedInit() ;
        document.forms[0][0].disabled = true ;
        document.forms[0][1].disabled = false ;
    }
    function DeInit()
    {
        VoiceObj.StgVoiceDeInit() ;
        document.forms[0][0].disabled = false ;

        document.forms[0][1].disabled = true ;
    }
</SCRIPT>
</HEAD>

<BODY>
<OBJECT ID="VoiceObj" CLASSID="CLSID:5FF0CA2F-A640-4F62-8EDF-C97F30B562FB">
    <PARAM NAME="_Version" VALUE="65536">
</OBJECT>
<P>Demo Voice with FlexTools - voiXtreme</P>
<FORM id="form1">
    <P><INPUT type="button" id="init" name="Init" onclick="javascript:Init();"
value="Init"></P>
    <P><INPUT type="button" id="deinit" name="DeInit"
onclick="javascript:DeInit();" value="DeInit"></P>
</FORM>
<A HREF="voixtreme2.htm">Go</A>
</BODY>
</HTML>
```

Fichier voixtreme2.htm

```
<HTML>
<HEAD>
<TITLE>Test page for Voice Object voiXtreme in FlexTools</TITLE>
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<META NAME="MobileOptimized" content="240">
<SCRIPT LANGUAGE="javascript">
    function OnSubmit()
    {
        VoiceObj.StgVoiceSpellSync( document.forms[0][0].value ) ;
        VoiceObj.StgVoiceSaySync( "Accepter, annuler ou finir?" ) ;

        VoiceObj.StgVoiceActivateRule("<commande>") ;
        VoiceObj.StgVoiceDeactivateRule("<digits>") ;
        VoiceObj.StgVoiceListenSync() ;
        commandsaid = VoiceObj.StgVoiceGetResultString() ;
        if ( commandsaid == "accepter" )
        {
            document.forms[0].submit();
        }
        else
        {
            document.forms[0][0].value = VoiceObj.StgVoiceGetResultString() ;
            if ( commandsaid != "finir" )
            {
                Listen() ;
            }
            else
            {
                VoiceObj.StgVoiceSaySync( "Merci. Au revoir." ) ;
            }
        }
    }
    function Listen()
    {
        VoiceObj.StgVoiceSaySync("quantité") ;
        VoiceObj.StgVoiceActivateRule("<digits>") ;
        VoiceObj.StgVoiceDeactivateRule("<commande>") ;
        VoiceObj.StgVoiceListenSync() ;
        document.forms[0][0].value = VoiceObj.StgVoiceGetResultString() ;
        OnSubmit() ;
    }
</SCRIPT>
</HEAD>

<BODY onload="javascript:Listen();" >
<OBJECT ID="VoiceObj" CLASSID="CLSID:5FF0CA2F-A640-4F62-8EDF-C97F30B562FB">
    <PARAM NAME="_Version" VALUE="65536">
</OBJECT>
    <P>Demo Voice with FlexTools</P>
    <FORM id="form1">
    <P>Quantity said: <INPUT type="text" id="qTTY" name="qTTY"
action="voixtreme2.htm"></P>
    <P><INPUT type="submit" onclick="javascript:OnSubmit();" value="Envier"></P>
    </FORM>
    <A HREF="voixtreme1.htm">Return</A>
</BODY>
</HTML>
```

Le programmeur devrait connaître la grammaire installée, que pour cet exemple est la suivante :

```
!grammar Digit;
!start <Speech>;

<Speech>: !repeat(<digits>,1,*) | <commande> ;

<digits>: 0 !id(0) |
          1 !id(1) |
          2 !id(2) |
          3 !id(3) |
          4 !id(4) |
          5 !id(5) |
          6 !id(6) |
          7 !id(7) |
          8 !id(8) |
          9 !id(9) ;

<commande> : accepter |
            annuler |
            commencer |
            finir ;
```