

Erreur n°1640 à la compilation d'un projet Visual FoxPro

Message : « Access to system Registry denied »

Message localisé : « Accès au registre système refusé »

A) Dans quel cas cette erreur est levée ?

B) Pourquoi ?

C) Quelles solutions de contournement ?

C.1) Désactiver UAC (/!\)

C.2) Lancer VFP avec élévation Administrateur

C.2.1) Modifier les propriétés de l'icône VFP9

C.2.2) Click droit lancer en tant qu'administrateur

C.2.3) Ruban explorer de Windows 8 « lancer en tant qu'administrateur »

C.3) Ajouter autorisation dans la base de registre

D) Retour « pourtant je n'ai pas défini de classe OLEPUBLIC... »

A) Dans quel cas cette erreur est levée ?

Les conditions suivantes doivent être réunies pour que la compilation d'un projet (commande BUILD) soit avortée en erreur 1640 :

A.1) Vous développez sur un système d'exploitation Windows Vista ou supérieur : Vista / Windows 2008 / Windows 7 / Windows 8 (que ce soit en version 32 ou 64 bits)

+

A.2) Le service UAC est activé.

UAC = User Acces Control (Control de compte de l'utilisateur).

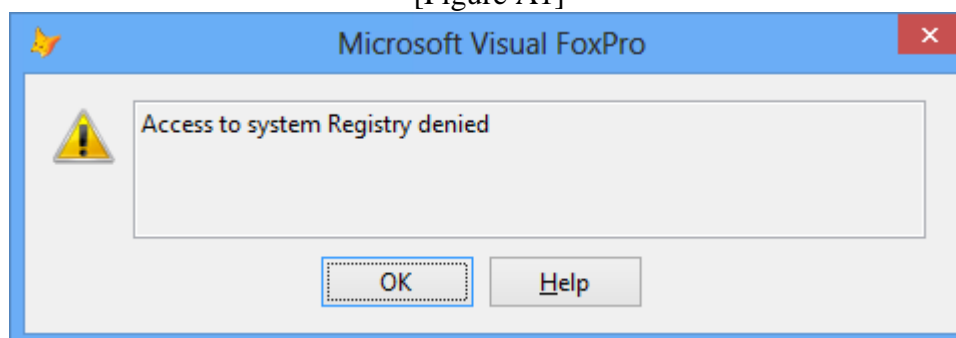
Celui-ci est activé par défaut sur ces systèmes cités ci-avant. (Et c'est une bonne chose !)

(Pour plus d'information généraliste sur UAC voir : http://fr.wikipedia.org/wiki/User_Account_Control)

+

A.3) Vous avez une ou plusieurs classes « OLEPUBLIC » dans votre projet

[Figure A1]



B) Pourquoi ?

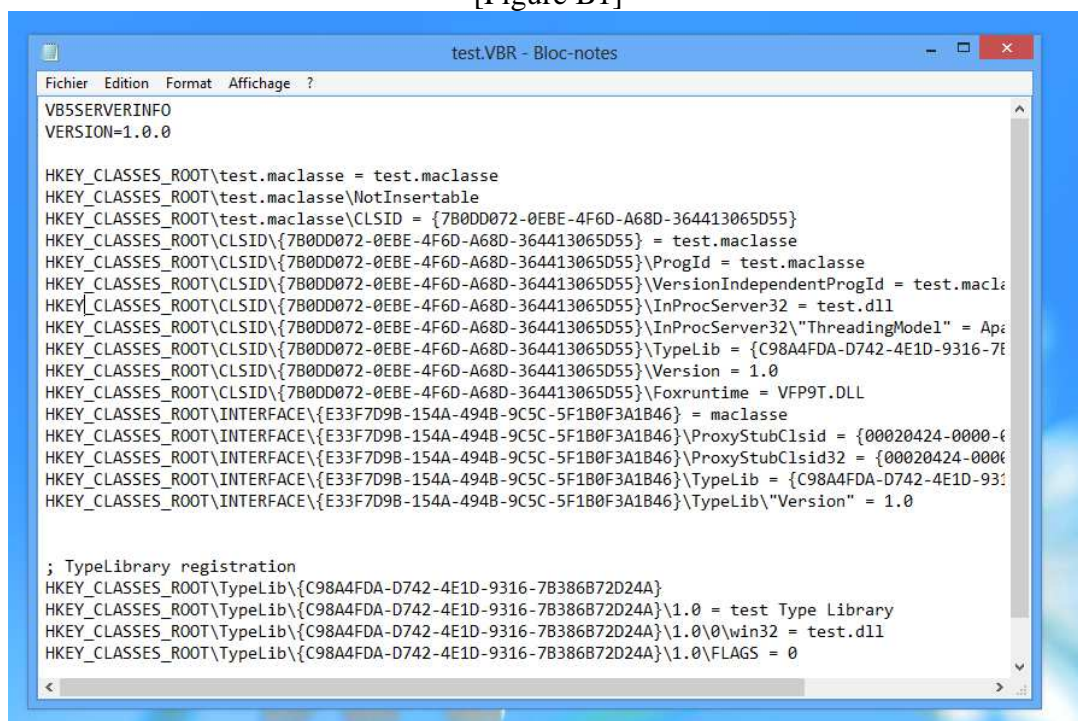
Lors de la compilation (que ce soit en EXE, DLL ou MTDLL) d'un projet contenant une classe ou plusieurs classes déclarées OLEPUBLIC (Classe publique = mécanisme des composants COM exposés dans Windows de génération « NT » qui en fait sa force) : le compilateur Visual FoxPro cherche à enregistrer la/les classes dans la base de registre.

La « fox team » n'a pas prévu la possibilité de dissocier le processus de compilation et celui de l'enregistrement en base de registre (à ma connaissance), ce qui est bien dommage dans le cas d'un développement sur une machine différente de l'utilisation (cas réseau).

Bref, la phase d'enregistrement des classes en base de registre nécessite le droit d'écriture sur l'entrée de ruche « HKEY_CLASSES_ROOT ».

Vous pouvez consulter les entrées nécessaires listées dans votre fichier « .VBR » lié à votre projet (éditable avec un éditeur de texte).

[Figure B1]



```
Fichier Edition Format Affichage ?
VB5SERVERINFO
VERSION=1.0.0

HKEY_CLASSES_ROOT\test.maclasse = test.maclasse
HKEY_CLASSES_ROOT\test.maclasse\NotInsertable
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55} = test.maclasse
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\ProgId = test.maclasse
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\VersionIndependentProgId = test.maclasse
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\InProcServer32 = test.dll
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\InProcServer32\ThreadingModel = Apartment
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\TypeLib = {C98A4FDA-D742-4E1D-9316-7B386B72D24A}
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\Version = 1.0
HKEY_CLASSES_ROOT\CLSID\{7B0DD072-0EBE-4F6D-A68D-364413065D55}\Foxruntime = VFP9T.DLL
HKEY_CLASSES_ROOT\INTERFACE\{E33F7D9B-154A-494B-9C5C-5F1B0F3A1B46} = maclasse
HKEY_CLASSES_ROOT\INTERFACE\{E33F7D9B-154A-494B-9C5C-5F1B0F3A1B46}\ProxyStubClsid = {00020424-0000-4000-0000-00000000}
HKEY_CLASSES_ROOT\INTERFACE\{E33F7D9B-154A-494B-9C5C-5F1B0F3A1B46}\ProxyStubClsid32 = {00020424-0000-4000-0000-00000000}
HKEY_CLASSES_ROOT\INTERFACE\{E33F7D9B-154A-494B-9C5C-5F1B0F3A1B46}\TypeLib = {C98A4FDA-D742-4E1D-9316-7B386B72D24A}
HKEY_CLASSES_ROOT\INTERFACE\{E33F7D9B-154A-494B-9C5C-5F1B0F3A1B46}\TypeLib\Version = 1.0

; TypeLibrary registration
HKEY_CLASSES_ROOT\TypeLib\{C98A4FDA-D742-4E1D-9316-7B386B72D24A}
HKEY_CLASSES_ROOT\TypeLib\{C98A4FDA-D742-4E1D-9316-7B386B72D24A}\1.0 = test Type Library
HKEY_CLASSES_ROOT\TypeLib\{C98A4FDA-D742-4E1D-9316-7B386B72D24A}\1.0\0\win32 = test.dll
HKEY_CLASSES_ROOT\TypeLib\{C98A4FDA-D742-4E1D-9316-7B386B72D24A}\1.0\FLAGS = 0
```

La ruche « HKEY_CLASSES_ROOT » n'étant pas virtualisée (contrairement à « HKEY_LOCAL_MACHINE »), l'écriture sans le droit d'élévation administrateur : l'exception 1640 est levée et la compilation est avortée (et ne génère pas le fichier EXE ou DLL !)

C) Quelles solutions de contournement ?

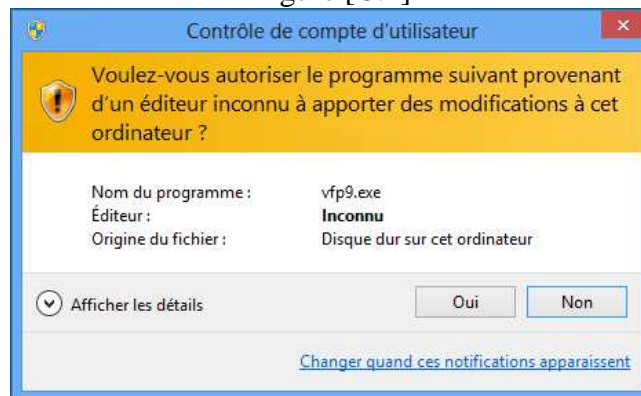
C.1) Désactiver UAC (/!\)

Désactiver la sécurité UAC n'est pas à recommander. Le processus d'élévation qui est maintenant intégré à Windows depuis la version « Vista » est un bienfait. Il est similaire au « su » des Unix ou « sudo » des Linux, cela nécessite pour l'utilisateur, même administrateur, de confirmer une élévation pour modifier le système, privant ainsi les « malwares » d'être invisibles contrairement aux systèmes XP et précédents.

C.2) Lancer VFP avec élévation Administrateur

Ces trois contournements sont équivalents : le but étant de lancer VFP avec une élévation.

Figure [C.2]



C.2.1) Modifier les propriétés de l'icône VFP9

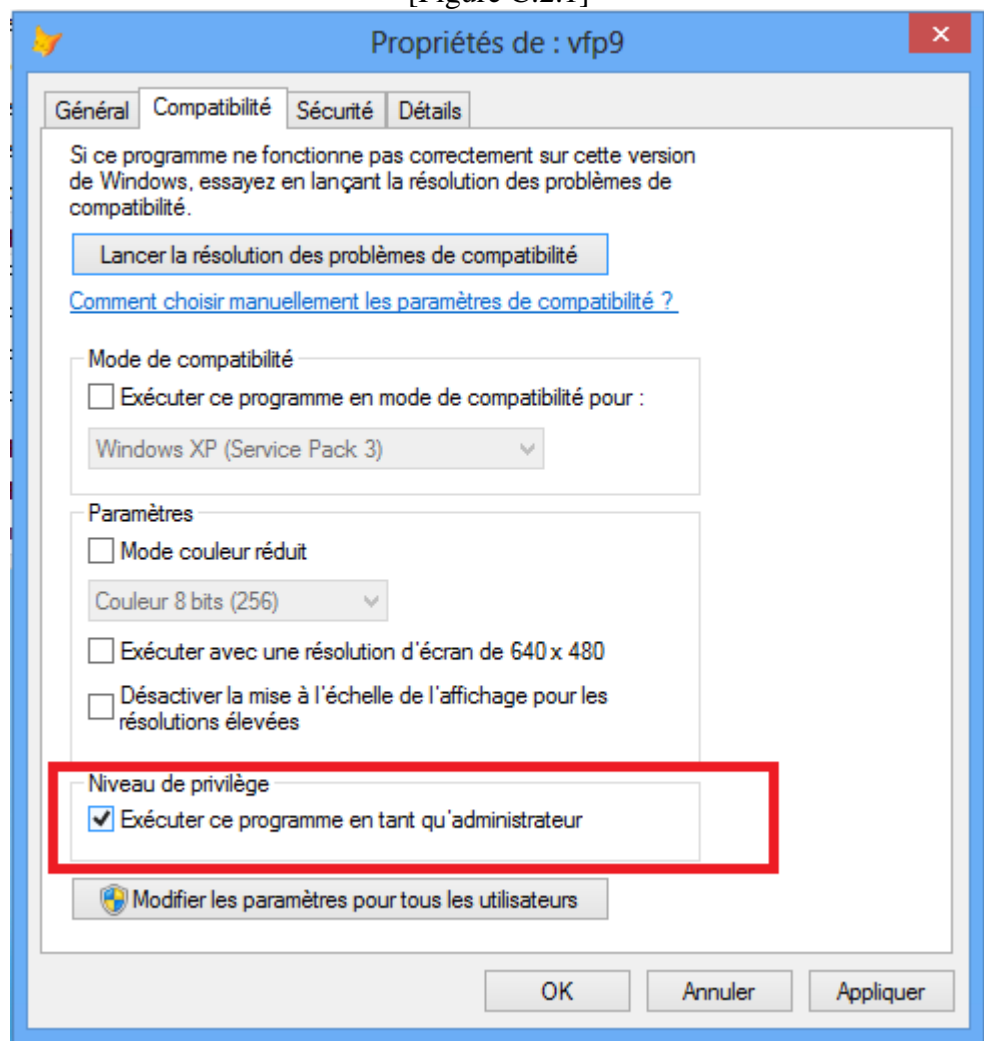
Cliquer « droit » sur l'icône de VFP -> Onglet « Compatibilité »

Puis, dans le « niveau de privilège » :

Cocher « Exécuter ce programme en tant qu'administrateur »

Cela aura pour conséquence d'élever systématiquement VFP9 en tant qu'administrateur à chaque lancement et le faire tourner dans un processus différent de celui de l'utilisateur.

[Figure C.2.1]

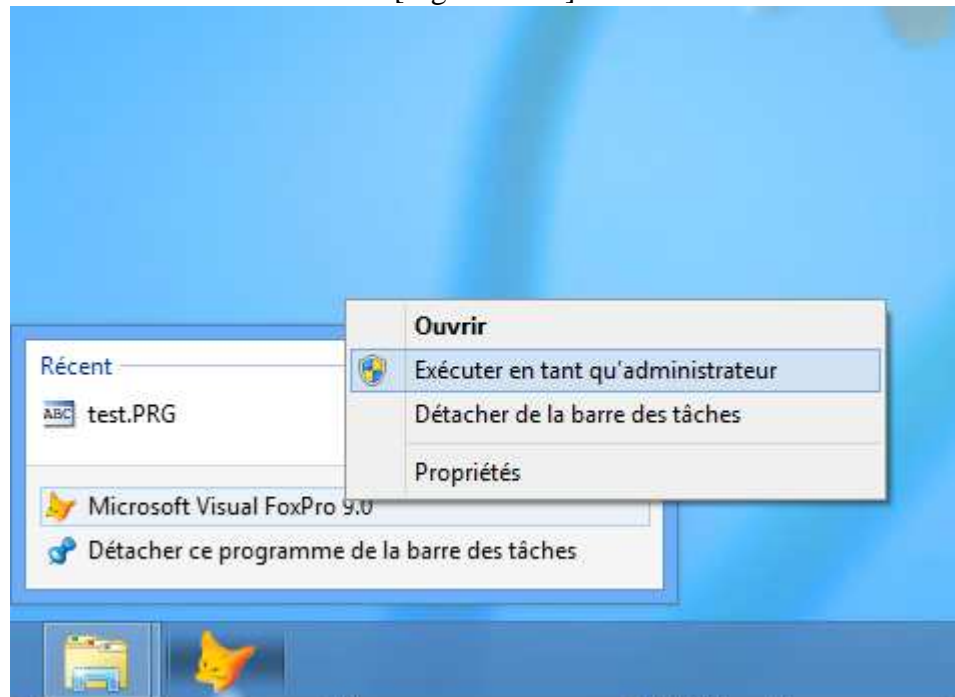


C.2.2) Click droit lancer en tant qu'administrateur

Cliquer « droit » sur l'icône de VFP -> « Exécuter en tant qu'administrateur »

Cela aura pour conséquence d'élever VFP9 mais uniquement le temps de ce processus.

[Figure C.2.2]

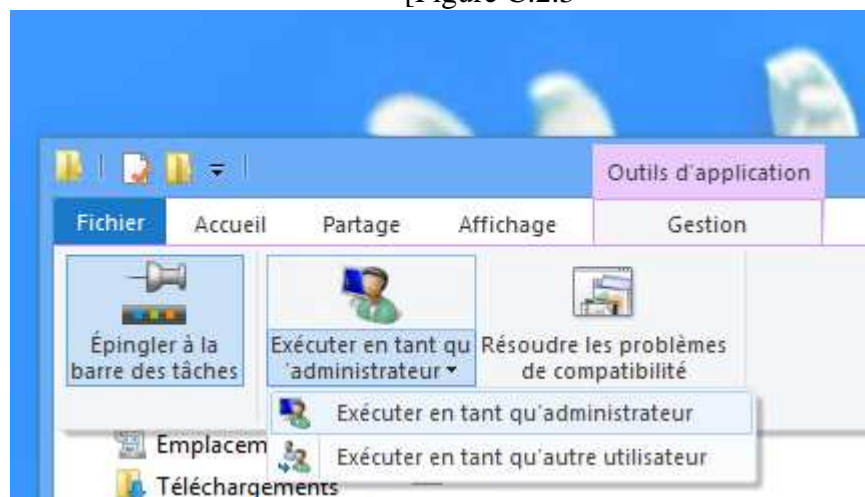


C.2.3) Ruban explorer de Windows 8 « lancer en tant qu'administrateur »

Le nouvel explorateur de Windows 8, dans son « ruban » de gestion permet de lancer en tant « qu'administrateur »

C'est équivalent à la solution §C2.2.2 ci-avant.

[Figure C.2.3]



C.3) Ajouter autorisation dans la base de registre.

Les inconvénients des solutions de contournement décrites ci avant en §C.2 :

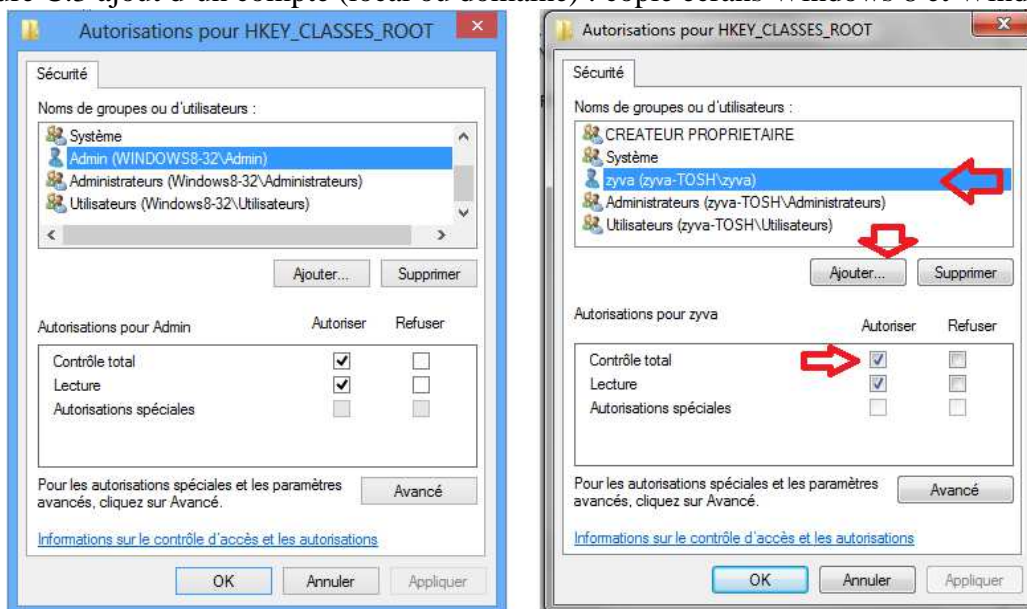
- A chaque lancement il faut confirmer l'élévation (éditeur inconnu...)
- Le développeur doit connaître le mot de passe administrateur local de sa machine de développement (ce qui n'est pas forcément le cas dans une entreprise ou TSE)
- Le développeur travaille en élévation administrateur, les tests de ses programmes ne seront pas forcément valides par rapport aux futurs utilisateurs du programme : En effet les accès aux répertoires protégés ne seront pas virtualisés comme pour un utilisateur, de même pour les accès à la ruche HLM de la base de registre, etc... donc des « comportements » différents sont possibles, voire passer à coté d'une erreur.
- Le fait de travailler en élévation : un processus différent est lancé, et l'environnement réseau n'est plus celui du développeur, les volumes réseaux mappés disparaissent, ce qui pose problème dans un environnement où le développeur utilise un compte de domaine sur Active Directory. Bien sûr il est possible de re-mapper les volumes réseaux après l'élévation mais c'est lourd...

La solution pour continuer à développer avec le compte du développeur (avec ou sans domaine), **sans recours à l'élévation**, mais permettre la compilation avec enregistrement dans la base de registre consiste à faire une « entorse »: autoriser le compte développeur à modifier HKEY_CLASSES_ROOT dans la base de registre : c'est une simple manipulation à faire avec REGEDIT :

Lancer REGEDIT, se placer sur HKEY_CLASSES_ROOT, click droit « autorisation » : **Ajouter** le compte domaine du développeur, ou le compte local du développeur, en ajoutant les droits d'écritures.

(On pourrait modifier le droit du groupe « utilisateurs » mais cela ouvre la modification de la base de registre à tous les utilisateurs : mieux vaut ajouter le compte particulier du développeur).

[Figure C.3 ajout d'un compte (local ou domaine) : copie écrans Windows 8 et Windows 7]



Personnellement, c'est cette dernière solution que nous avons préféré, car les développeurs travaillent ainsi sans élévation, avec un compte en domaine AD, préservant ainsi les unités réseaux montées et les tests des applications « comme » un utilisateur. (excepté l'accès HK_ROOT)

D) Retour : « pourtant je n'ai pas défini de classe OLEPUBLIC... »

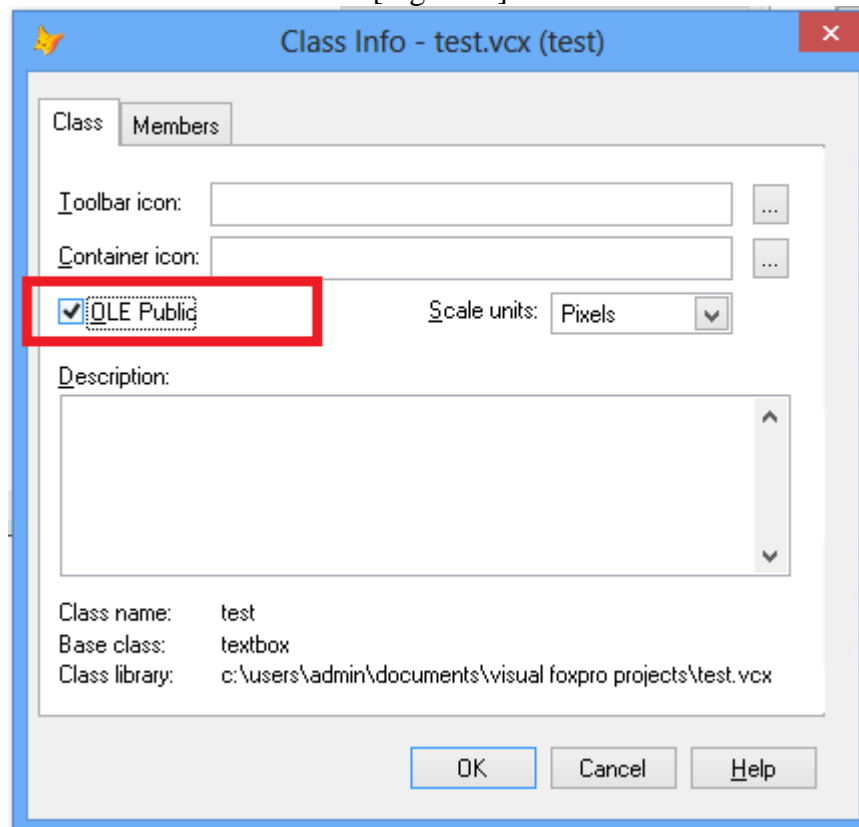
Une erreur 1640 c'est que forcement qu'il y a au moins une classe OLEPUBLIC dans le projet et VFP cherche à l'enregistrer.

- Cette classe peut être en programme (PRG) et contient le mot clé « OLEPUBLIC » :

```
DEFINE CLASS NOMde laClasse as BaseClass OLEPUBLIC  
ENDDDEFINE
```

- Mais cela peut être aussi une classe visuelle qui a été cochée « OLEPUBLIC »

[Figure D]



Cordialement
Francis FAURE