

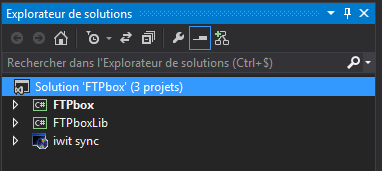
iwit sync

|  |
| --- |
| François Garcia  23/06/2015 |

Guide d’utilisation expert

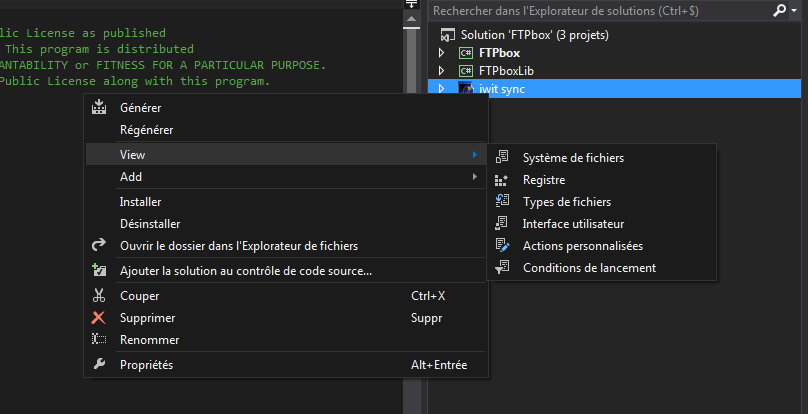
Mise à jour du code

## DEFINITION DES CLASSES



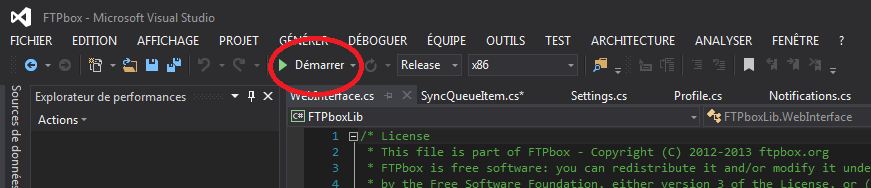
Nous disposons ici de trois éléments constituants notre solution.

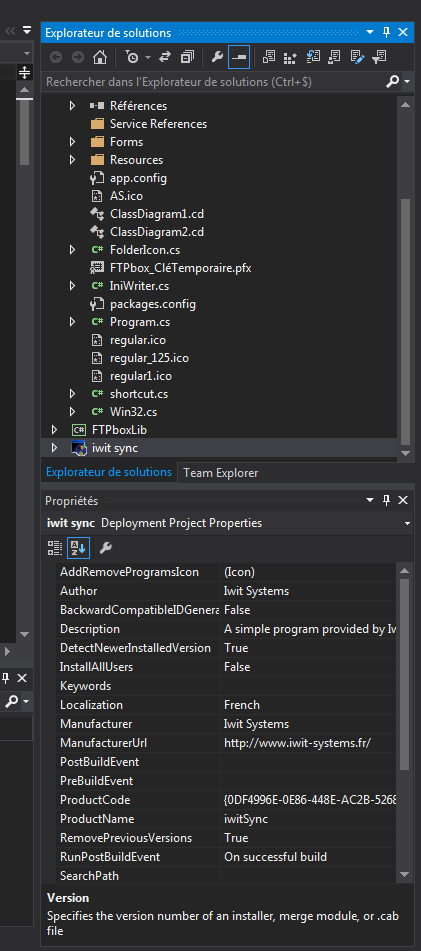
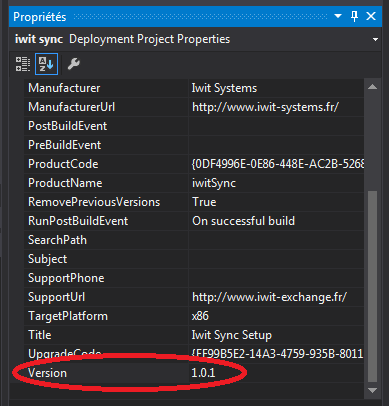
* FTPbox : Contiens les paramètres propres à la configuration d’usine de l’application ainsi que les ressources d’images. Ce projet contient également tout ce qui est relatif à l’interface graphique que l’utilisateur peut voir tel que « fMain.cs » correspondant au menu de configuration ou encore « Setup.cs » servant à mettre en place la configuration lors de la première utilisation de l’application.
* FTPboxLib : Contiens toutes les classes permettant le bon fonctionnement de l’application. Il s’agit des classes maitresses permettant la synchronisation etc.
* ***« AccountController.cs »*** : Permet de réguler toutes les autres classes du projet FTPboxLib.
* ***« AESEncryption.cs »*** : Permet de gérer le cryptage tel que celui du mot de passe contenu dans le fichier de configuration
* ***« Client.cs »*** : Pilier de la communication avec le serveur, la classe combine les librairies de FTP et de SFTP.
* ***« ClientItem.cs »*** : Représente un objet (fichier, dossier) trouvé sur le serveur. Elle est utilisée par la classe « Client.cs ».
* ***« Common.cs »*** : Classe permettant d’utiliser les méthodes et propriétés qui n’appartiennent pas à un profil spécifique.
* ***« DecryptionInfo.cs »*** : Infos nécessaires pour crypter et décrypter le mot de passe du fichier de config. Supprimé du code source pour des soucis de sécurité.
* ***« Enums.cs »*** : Contient toutes les énumérations utilisés pour voir et partage les objets récent.
* ***« EventArgs.cs »*** : Utilisé pour afficher les barres de progression etc.
* ***« Extensions.cs »*** : Collection d’extension permettant de remplacer les slashs en anti-slashs (par exemple).
* ***« FileLog.cs »*** : Gère la lecture et écriture du fichier ou dossier de log.
* ***« FileLogItem.cs »*** : Utilisé par la classe « FileLog.cs » permet de stocker dans un fichier du chemin d’accès principal le « temps de la dernière écriture » des dossiers à distance et local.
* ***« FolderWatcher.cs »*** : Surveille tous les changements dans le dossier local et ajouter ces éventuels changement dans la queue de synchronisation.
* ***« IgnoredList.cs »*** : Identifie les objets ignorés en se basant  sur les préférences de l’utilisateur.
* ***« Notifications.cs »*** : Traduis n’importe quelles notifications et les envoie vers la classe principale « fMain.cs » afin qu’elles soient affichées.
* ***« Profile.cs »*** : Classe permettant de stocker les informations d’un seul profil utilisateur.
* ***« Settings.cs »*** : Permet d’utiliser le chargement et la sauvegarde des préférences utilisateurs provenant du fichier de configuration.
* ***« SyncQueue.cs »*** : Gestion des objets à synchroniser en établissant une queue de synchronisation.
* ***« SyncQueueItem »*** : Utilisé par la classe « SyncQueue », stock toutes les données de chaque objets contenu dans la queue de synchronisation.
* ***« Translation.cs »*** : Permet de gérer toutes les traductions pour l’interface de l’utilisateur (inutile dans notre cas car utilisation forcé).
* ***« WebInterface.cs »*** : Classe inutile également dans notre système, permet la gestion de tout ce qui peut être relative à l’utilisation de l’interface web (important pour vous plus tard).
* iwit sync : Contient le projet dans sa globalité afin d’installer les ressources sur un ordinateur plus tard. Ce projet permet la sélection individuelle de chaque fichier et leur comportement, offre également la possibilité de choisir une plateforme de déploiement tel qu’un déploiement en x86 ou en x64 ou en multiple. Pour accéder aux paramètres relatifs à l’installateur il suffit de faire clic droit sur le projet et d’aller dans le menu « view »



## COMPILATION – MISE A JOUR

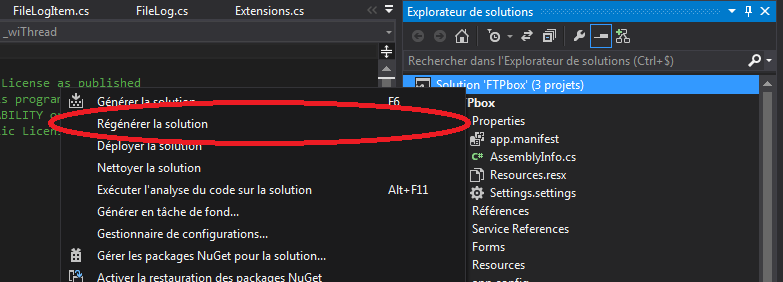
Pour vérifier qu’aucune coquille ne se soit glissé dans le code lors d’une manipulation il est conseillé de compiler le code avant de déployer la solution vers un exécutable. Pour faire ceci il suffit de cliquer sur « Démarrer » se trouvant sur la barre principale de l’application.



Si toutefois aucun erreur ne parait présente l’application est prête à être déployer. Afin de correctement déployer l’application il est important de souligner le fait qu’il s’agira d’une mise à jour. Pour faire ceci il faut se rendre dans les propriétés du projet en cliquant une fois sur le projet « iwit sync ». Vous devriez voir apparaître ceci.

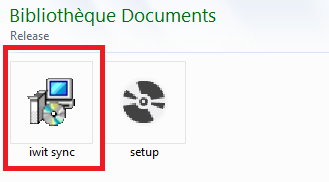
Une fois le menu ci-dessous ouvert il nous suffit de nous rendre sur version et d’incrémenter votre numéral. La version actuelle est « 1.0.1 » ceci permet au programme de détecter qu’une version antérieur ce trouvé sur les machines et permet donc d’utiliser sa fonction « RemovePreviousVersions ».

Lorsque toutes ces modifications ont été appliquées vous pouvez finir par faire un clic droit sur la solution « FTPbox  (3 projets) » et « Régénérer la solution ».



Un installateur va alors être généré au sein de votre dossier du code source au path suivant :

*« FTPbox-master\Windows\iwit sync\Release »*

Il ne vous reste plus qu’à déployer votre projet sur les machines clientes et l’installation se déroulera comme n’importe quelle autre installation d’application normale.

Sur les deux fichiers ne prendre seulement « iwit sync »

Tour de l’application

## FONCTIONNEMENT GLOBAL

**UTILISATEUR**

Le fonctionnement de l’application se base sur une installation simple et un lancement par la suite automatique après le premier lancement de l’application. L’utilisateur n’a à sa disposition qu’un accès limité à la configuration de l’application.

L’utilisateur ne peut accéder qu’aux paramètres généraux tels que le choix de faire apparaitre ou non les notifications. La possibilité de choisir sa fréquence de synchronisation, et ainsi de pouvoir basculer entre une synchronisation manuelle ou bien automatique avec le choix d’un temps de synchronisation.

L’utilisateur est également libre de modifier la vitesse de téléchargement ou d’upload par le biais de deux formulaires ou la valeur par défaut est 0.

Il est finalement possible pour l’utilisateur de modifier les paramètres de son compte tel que l’adresse du serveur ou son identifiant/mot de passe et également de Recréer un raccourci sur le bureau si l’utilisateur l’avait supprimé auparavant.

**MAINTENANCE IWIT**

Pour finir sur l’utilisation globale, il est mis à disposition des administrateurs de l’application le raccourci suivant : CTRL + SHIFT + F afin de faire apparaitre le contenu dans la tab maintenance dans l’espace configuration de l’application.

Ici nous trouverons divers outils tel que la possibilité de sélectionner la direction de la synchronisation mais encore de pouvoir configurer plusieurs compte et de switcher entre eux ou d’en ajouter/supprimer et d’afficher un récapitulatif pour chaque compte.

Il est également possible d’établir un filtrage comme une synchronisation sélective permettant de cocher ou non les éléments à synchroniser dans le dossier de synchronisation. Il est également possible d’ignorer des extensions tel que les .jpeg par exemple et aussi modifier le préfixe des fichiers temporaires (les fichiers temporaires interviennent lorsque l’application télécharge un fichier, cela lui permet de d’abord télécharge le fichier temporaires utilisant le préfixe de votre choix et par la suite transforme ce fichier temporaire en fichier normal. Cela évite tout problème lié à la synchronisation).

En dehors des filtres l’application permet également de choisir si l’application doit se lancer au démarrage de Windows, si on doit activer l’authentification, c’est-à-dire la demande systématique du combo identifiant/mot de passe à chaque démarrage de Windows et également de faire apparaitre le fichier des logs.

## IMPACT SUR LE SYSTEME

Lors de la première configuration et du lancement de certaines options par l’utilisateur ou par l’administrateur de l’application iwit, le programme créera des entrées dans le registre et également des entrées dans d’autre dossier tel que le dossier AppData.

## LOCALISATION DES FICHIERS DE CONFIGURATION – REGISTRE

"But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master-builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful. Nor again is there anyone who loves or pursues or desires to obtain pain of itself, because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to obtain some advantage from it? But who has any right to find fault with a man who chooses to enjoy a pleasure that has no annoying consequences, or one who avoids a pain that produces no resultant pleasure?"

Fonctions avancées

## UTILISATION DU MODE MAINTENANCE