/

Guide de mise en route

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Commentaire** | **Auteur** | **Société** |
| 0.1 | 22/06/2017 | Création du document. | Vincent JACQUOT | EID |
| 0.2 | 02/08/2017 | Artifactory (NuGet) et SQL Developer. | Vincent JACQUOT | EID |
| 0.3 | 10/08/2017 | Connexion à TFS | Vincent JACQUOT | EID |
| 0.4 | 16/08/2017 | Connexion à une BDD, chiffrement | Vincent JACQUOT | EID |
| 0.5 | 04/09/2017 | Précisions sur la base et le driver Oracle | Vincent JACQUOT | EID |
| 0.6 | 11/09/2017 | Clé de registre pour la base de données | Vincent JACQUOT | EID |
| 0.7 | 12/09/2017 | Fichier de connexions SQL Develop | Vincent JACQUOT | EID |
| 0.8 | 22/09/2017 | Mise à jour des connexions SQL | Vincent JACQUOT | EID |
| 0.9 | 25/09/2017 | Mise à jour pour passage en 64 bit | Vincent JACQUOT | EID |
| 0.10 | 11/09/2017 | Correction du fichier des corrections | Vincent JACQUOT | EID |
| 1.0 | 11/09/2017 | Nom DNS pour TFS | Vincent JACQUOT | EID |
| 1.1 | 12/09/2017 | Correction des URL dans les captures | Vincent JACQUOT | EID |
| 1.2 | 13/09/2017 | Précision sur IIS Express 64 bit | Vincent JACQUOT | EID |
| 1.3 | 23/10/2017 | Clés de registre compatibles 32 et 64 bit | Vincent JACQUOT | EID |
| 1.4 | 16/10/2017 | Clés de registre spécifiques nouveau framework | Vincent JACQUOT | EID |
| 1.5 | 02/11/2017 | URL de Sonar et des Web Services | Vincent JACQUOT | EID |
| 1.6 | 03/11/2017 | Changement de l’URL du moteur sur NI | Vincent JACQUOT | EID |
| 1.7 | 06/11/2017 | Procédure de lancement des batchs en local et prise en compte du caractère monolithique du moteur | Vincent JACQUOT | EID |

Table des matières

[1 Base de registre 3](#_Toc497724080)

[2 Oracle 3](#_Toc497724081)

[2.1 Driver Oracle 3](#_Toc497724082)

[2.2 SQL Developer 3](#_Toc497724083)

[2.2.1 Résolution du problème de démarrage 4](#_Toc497724084)

[2.2.2 Connexion aux bases de données 4](#_Toc497724085)

[3 Visual Studio 6](#_Toc497724086)

[3.1 Dépôt NuGet 6](#_Toc497724087)

[3.2 Connexion à Team Foundation Server 8](#_Toc497724088)

[3.3 Plateforme de IIS Express 9](#_Toc497724089)

[3.3.1 Moteur 9](#_Toc497724090)

[3.3.2 DevBooster 9](#_Toc497724091)

[4 Développement sur l’application du moteur 10](#_Toc497724092)

[4.1 Récupérer la solution depuis Git 10](#_Toc497724093)

[4.2 Démarrer la solution sur son poste de développement 12](#_Toc497724094)

[4.3 Appeler un Web Service sur son poste de développement 14](#_Toc497724095)

[5 Développement sur une application batch 16](#_Toc497724096)

[5.1 Bases de données 16](#_Toc497724097)

[5.2 Dépendances techniques 17](#_Toc497724098)

[5.3 Lancer un batch sur son poste de développement 17](#_Toc497724099)

[6 Liens utiles 19](#_Toc497724100)

# Base de registre

Une partie de la configuration nécessaire aux développements est stockée en base de registre. Pour pouvoir exécuter les applications du moteur sur un poste de développement, une configuration minimale doit être insérée dans la base de registre :

* La configuration du chiffrement
* Les chaînes de connexion des bases de données de test

Pour appliquer cette configuration, il faut inscrire les fichiers suivants dans la base de registre :

* BaseDeDonnees.reg
* Chiffrement.reg

Les fichiers sont disponibles ci-dessous :



# Oracle

Les briques du moteur et les batchs utilisent une base Oracle en version 11g pour gérer la persistance. La base de données de l’environnement de test est accessible depuis les postes de développement via les applications de test ou le client SQL Developer.

## Driver Oracle

Pour accéder à une base de données Oracle depuis l’environnement d’exécution .NET, certains programmes nécessitent que le drive Oracle pour .NET en version 11.2.0.3.0 soit installé sur la machine. L’installation du driver Oracle nécessite d’avoir les droits administrateurs, elle doit être demandée aux internes EID qui transmettrons la demande au STU qui effectuera l’installation.

Avant de d’effectuer ou de demander l’installation du driver Oracle, vérifiez que les applications sur lesquelles vous travaillez ne peuvent pas se connecter à la base Oracle sans le driver. A l’heure actuelle, seules les applications suivantes nécessitent l’installation du driver Oracle :

* Les briques moteur basées sur l’*ancien framework* (ASMX et NHibernate).
* Les batchs de traitement de masse qui opèrent directement en base de données.

## SQL Developer

SQL Developer est l’outil utilisé pour manipuler les bases de données de développement. Il est présent sur les machines dans le répertoire C:/sqldeveloper/sqldeveloper.exe.

### Résolution du problème de démarrage

Il y a actuellement un problème sur les VM qui provoque une erreur au démarrage de SQL Developer. Au démarrage du programme, SQL Developer indique qu’il ne trouve pas la librairie MSVCR71.dll.

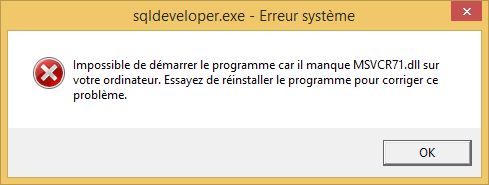


Figure - Erreur indiquée au démarrage de SQL Developer

Cette DLL est en fait disponible dans le répertoire C:\sqldeveloper\jdk\jre\bin. La manière la plus simple de corriger ce problème est de copier le fichier msvcr71.dll à la racine du répertoire C:\sqldeveloper. Une fois le fichier en place, SQL Developer accepte de démarrer.

### Connexion aux bases de données

Pour se connecter à une base de données, il faut ajouter les nouvelles connexions dans le panneau ***Connexions*** à gauche de l’espace de travail. Le fichier ci-dessous permet d’importer toutes les connexions dont vous avez besoin pour développer.



Faites un clic-droit sur la racine de la liste des connexions pour ouvrir le menu contextuel puis cliquez sur ***Importer des connexions…***

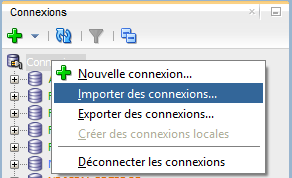


Figure – Import de connexions dans SQL Developer

Vous devez sélectionner le fichier XML qui contient les connexions puis cliquez sur ***suivant***.

L’étape suivante vous permet de renseigner la clé de cryptage qui protège les mots de passe stockés dans le fichier. Le fichier contient uniquement les mots de passe des bases de données de développement. Le mot de passe à renseigner est EIF\_BASES.

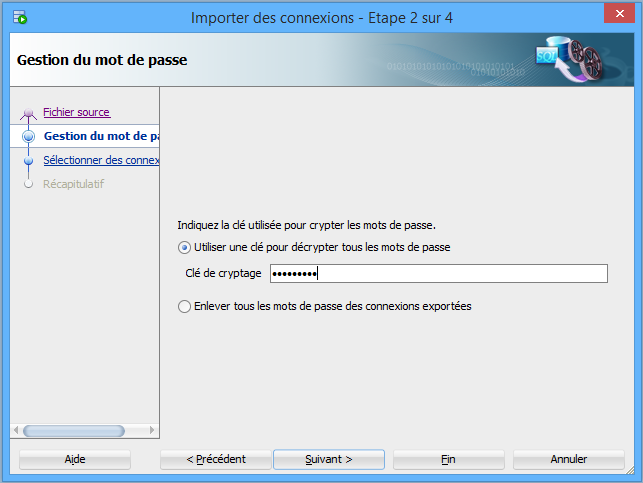


Figure - Saisie de la clé de cryptage des mots de passe

Sélectionnez enfin les connexions que vous souhaitez importer. A la fin de l’import, les nouvelles connexions apparaissent dans la liste des connexions à droite de l’espace de travail. Pour vous connecter, double-cliquez sur le nom de connexion. SQL Developer ouvre alors une fenêtre qui permet d’écrire et d’exécuter des requêtes.

Il est possible de configurer individuellement les connexions importées. Par exemple, si la configuration par défaut ne vous convient pas ou que le mot de passe n’est pas renseigné. Pour cela, faites un clic droit sur la connexion et cliquez sur ***Propriétés***. La fenêtre qui s’ouvre vous permet de configurer la connexion.

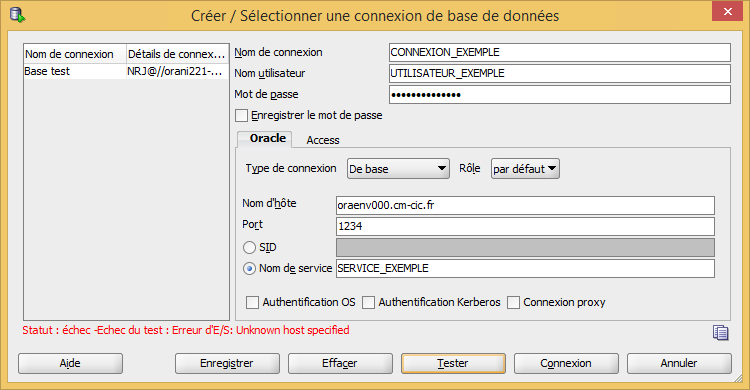


Figure - Configuration d'une connexion dans SQL Developer

Cliquez sur le bouton ***Tester*** pour vérifier que les informations sont valides et que la connexion fonctionne. Si la connexion fonctionne, cliquez sur le bouton ***Enregistrer***.

# Visual Studio

## Dépôt NuGet

Les différentes briques du moteur utilisent des paquets NuGet pour référencer les briques communes et les références partagées. Les développeurs EIT disposent d’un dépôt NuGet spécifique hébergé sur *Artifactory*. Il contient les DLL partagées internes et sert également de proxy pour dépôt le NuGet officiel qui contient l’outillage de Microsoft (Entity Framework, MEF).

Le dépôt NuGet doit être configuré dans Visual Studio. La configuration est accessible depuis le menu ***Tools***, ***NuGet Package Manager*** puis ***Package Manager Settings***.

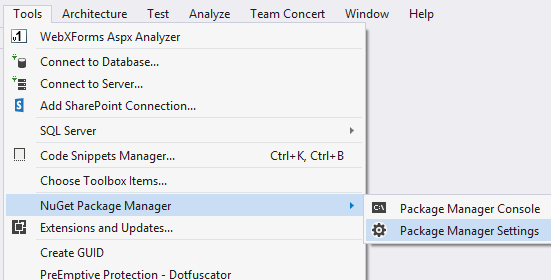


Figure - La configuration des dépôts NuGet est accessible depuis le menu *Tools*

La section ***Package Sources*** contient la liste des dépôts configurés dans Visual Studio. Le dépôt *Artifactory* doit y être ajouté et placé en première position dans la liste de priorité. Son URL est la suivante : <http://artifactory.eit.cm-cic.fr:8081/artifactory/api/nuget/nuget>.

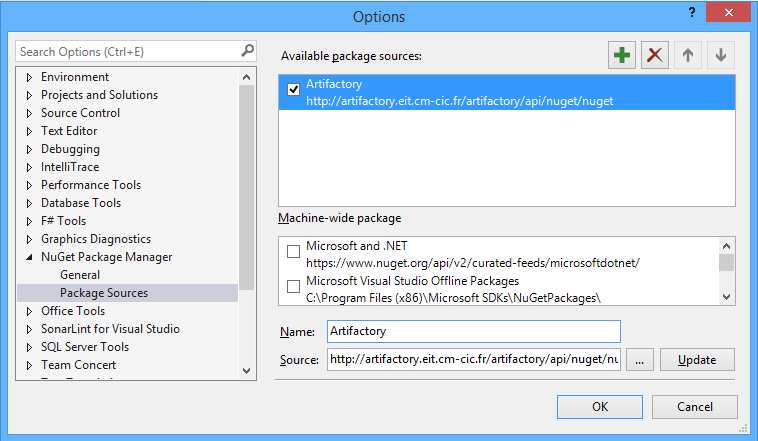


Figure - Configuration du dépôt Artifactory

Les autres dépôts peuvent être désactivés, en particulier si la machine de développement n’a pas accès à Internet.

## Connexion à Team Foundation Server

Team Foundation Server (TFS) est notre plateforme d’intégration pour les applications du moteur. Il sert également de contrôleur de code source puisqu’il héberge les dépôts Git des applications.

Pour connecter Visual Studio à TFS, rendez-vous dans la section ***Connect*** de la fenêtre ***Team Explorer***. Cliquez ensuite sur le menu ***Manage Connections*** puis l’entée ***Connect to Team Project***.

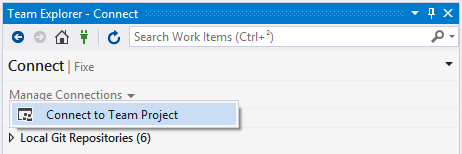


Figure - Accès à la configuration des serveurs TFS depuis le *Team Explorer*

La première fois que vous configurez une connexion à TFS, la liste des serveurs sera vide. Accédez à la configuration de la liste des serveurs en cliquant sur le bouton ***Servers…*** Ajoutez un nouveau serveur en renseignant l’URL suivante : <http://tfs.eit.cm-cic.fr:8080/tfs> et en validant en cliquant sur OK.

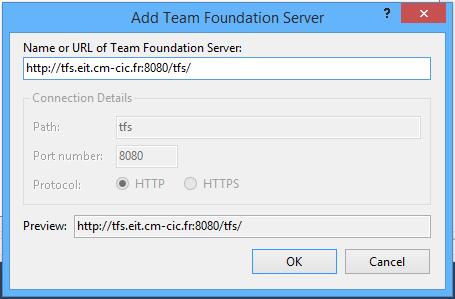


Figure - Configuration du nouveau serveur

Une fois le nouveau serveur ajouté, il faut sélectionner la **Collection** et le **Projet** adéquats. Les collections et les projets correspondent à deux niveaux d’organisation dans TFS, une collection étant un regroupement de projet. Vous devez sélectionner la collection **EITCollection** puis cocher le projet ***Fixe***.

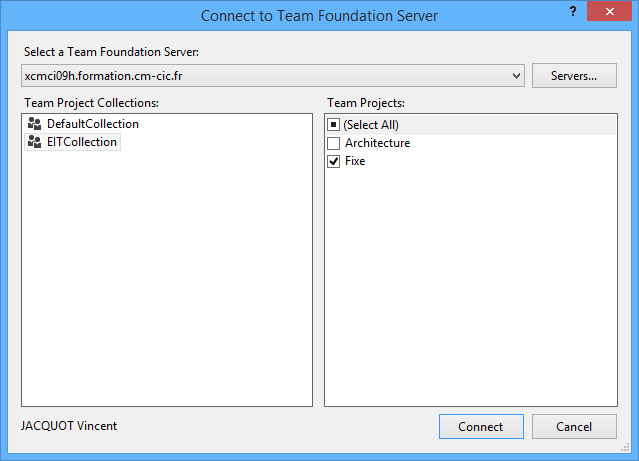


Figure - Sélections des collections et des projets sur le serveur TFS

Une fois ces étapes réalisées, votre *Team Explorer* sera connecté aux projets TFS sélectionnés. Vous aurez accès aux dépôts Git et aux éléments de travail des projets.

## Plateforme de IIS Express

### Moteur

Pour exécuter les assemblages du framework EIT (le moteur), Visual Studio 2015 doit être configuré pour exécuter la version 64 bit d’IIS Express.

Dans Visual Studio, ouvrez les options à partir du menu ***Tools***. Dans les options, ouvrez la sous-section ***Web Projects*** dans la section ***Projects and Solutions***. Cochez la case ***Use the 64 bit version of IIS Express for web site and projects***. Fermez les options en cliquant sur le bouton ***OK*** pour valider.

### DevBooster

Certains assemblages requis par DevBooster ne fonctionnement pas en 64 bit, il faut donc suivre la procédure inverse pour exécuter les IHM DevBooster.

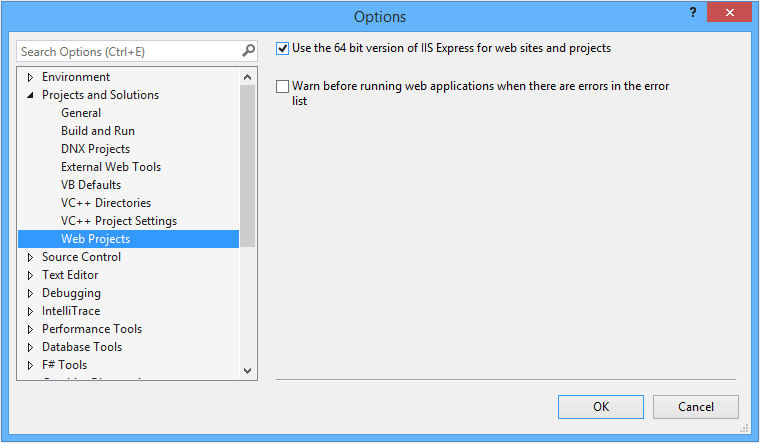


Figure – Changement de l’architecture d'IIS Express (32 ou 64 bit)

# Développement sur l’application du moteur

## Récupérer la solution depuis Git

La solutions Visual Studio de l’application du moteur est versionnée dans le dépôts Git EIT.Fixe. Pour récupérer une solution, il faut cloner le dépôt Git *EIT.Fixe* distant sur le poste local à partir de Visual Studio 2015.

Une fois que vous avez connecté votre Visual Studio au projet *Fixe* du serveur TFS, vous retrouvez la liste des dépôts dans la section ***Connect*** de la fenêtre ***Team Explorer***. Faites un clic-droit sur le dépôt *EIT.Fixe* qui se trouve dans le projet *Fixe*, puis cliquez sur ***Clone…***

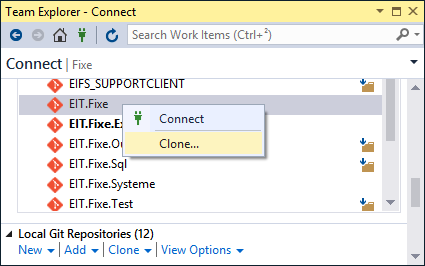


Figure - Cloner un dépôt depuis un projets TFS

Le formulaire de clonage sera alors automatiquement pré-rempli. Changez le répertoire pour placer le dépôt dans C:\TFS\EIT.Fixe. Assurez-vous que l’option ***Recursively Clone Submodules*** est cochée puis cliquez du **Clone**.

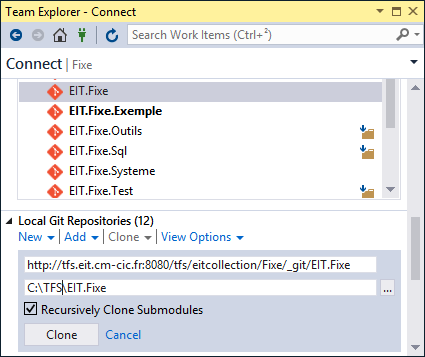


Figure - Formulaire de clonage d'un dépôt

Une fois le dépôt chargé, la liste des solutions Visual Studio existantes à la racine du dépôt est affichée dans la section *Home*. La solution EIT.Fixe contient tous les projets. Les autres solutions sont spécifiques à chaque domaine et ne contiennent que les projets appartenant au domaine. Pour ouvrir une solution, double-cliquez sur la ligne correspondante.

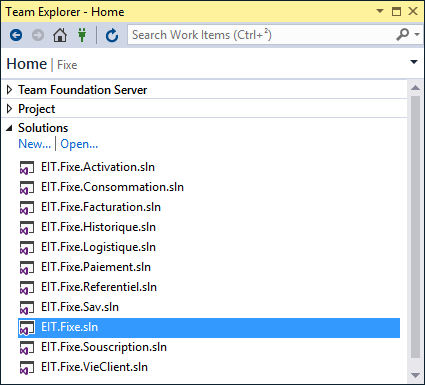


Figure - Section *Home* du *Team Explorer* avec la liste des solutions du dépôt EIT.Fixe

Une fois la solution chargée, n’oubliez pas de vous placer sur la bonne branche du dépôt avant de commencer à coder.

## Démarrer la solution sur son poste de développement

Seule la solution *EIT.Fixe* peut être exécutée comme une application Web. Pour démarrer la solution, assurez-vous que le projet ***EIT.Fixe.Infrastructure.WebApplication*** est défini comme projet de démarrage. Le projet *EIT.Fixe.Infrastructre.WebApplication* devrait apparaître en gras dans la liste des projets de la solution. Si ce n’est pas le cas, faites un clic droit sur le projet puis sur ***Set as StartUp Project*** dans le menu qui apparaît.

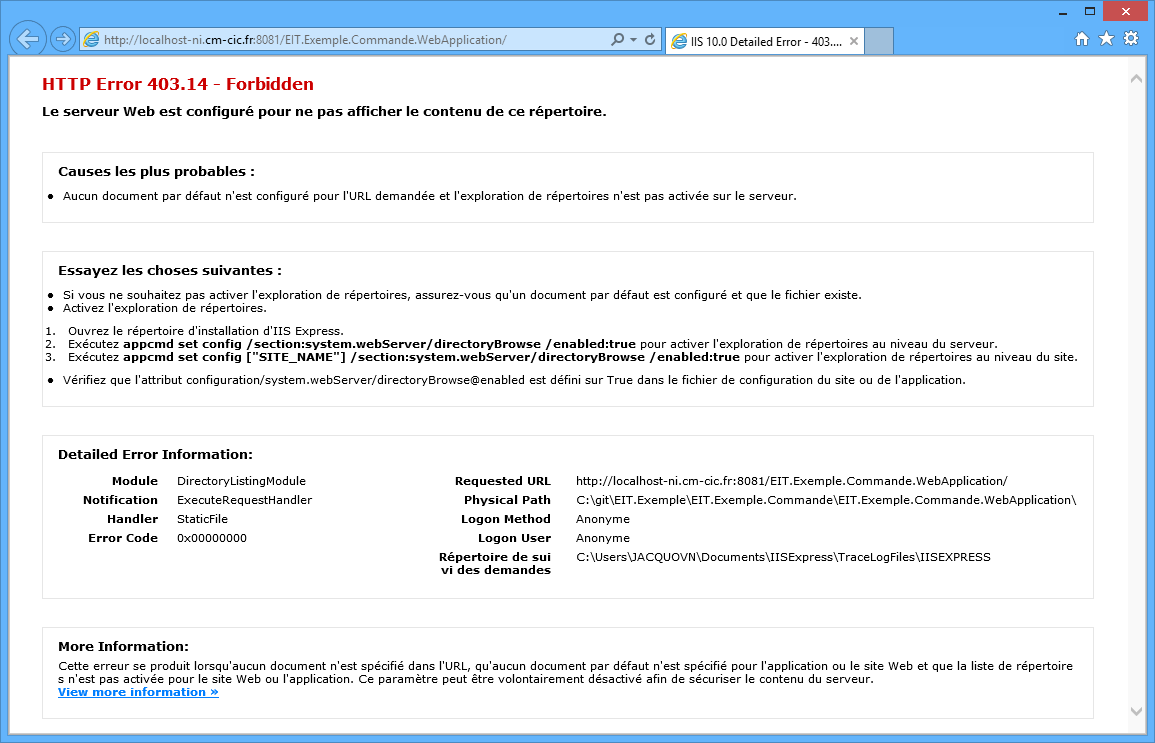


Figure - Une erreur 403 apparaît au lancement de la solution

Lorsque vous lancez la solution, Visual Studio va utiliser IIS Express pour héberger l’application web. Lorsque le navigateur apparaît, une page vous indique une erreur HTTP 403. Cette erreur est normale car l’application ne sert aucune page sur l’URL racine, IIS Express vous refuse donc l’accès.

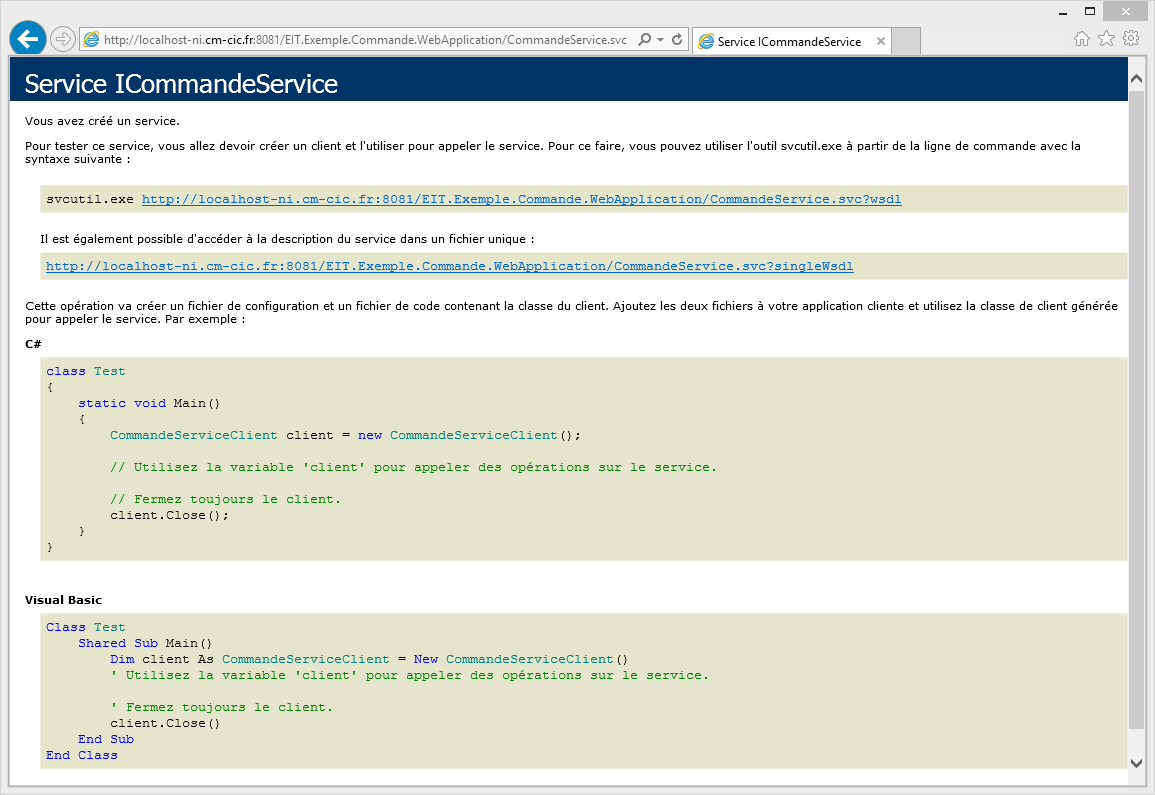


Figure - Page de description du service *CommandeService*

Pour vérifier le fonctionnement, ajoutez à l’URL le nom de l’un des services publiés par l’application suivi de l’extension **.*svc***. Par exemple, la page correspondant au service du contrat *ICommandeService* est disponible à l’URL *CommandeService.svc*. Vous devriez alors obtenir une page qui contient les liens vers les fichiers *WSDL* qui décrivent le service.

## Appeler un Web Service sur son poste de développement

Pour appeler un Web Service manuellement sur un poste de développement, il faut disposer d’un client comme *SoapUI* ou *WCF Test Client*. ***WCF Test Client*** est mis à disposition avec Visual Studio. Il permet de générer un client pour appeler un Web Service à partir d’un fichier *WSDL*. Vous pouvez le trouver sur votre machine aux emplacements suivants :

* C:\Program Files (x86)\Microsoft Visual Studio 14.0\Common7\IDE\WcfTestClient.exe
* C:\Program Files (x86)\Microsoft Visual Studio 11.0\Common7\IDE\WcfTestClient.exe

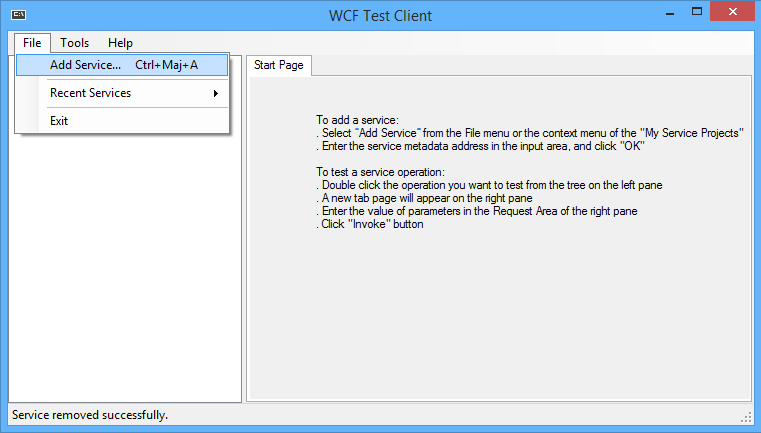


Figure - Ajouter un service dans *WCF Test Client*

Après voir ouvert WCF Test Client, vous pouvez générer un client pour appeler un service en cliquant sur ***Add Service*** dans le menu ***File*** ou avec le raccourci ***Ctrl+Maj+A***.

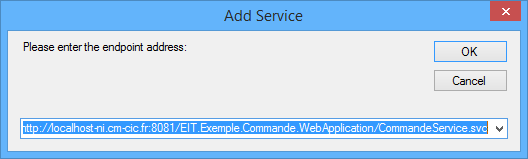


Figure - Renseigner l'URL du service pour générer un client

Il vous faut alors renseigner l’adresse du service (c’est-à-dire la même que la page de description du service) ou directement l’adresse du *WSDL*. Voir [la section précédente](#_Démarrer_une_solution) pour savoir comment obtenir ces adresses. Cliquez ensuite sur OK pour générer le client.

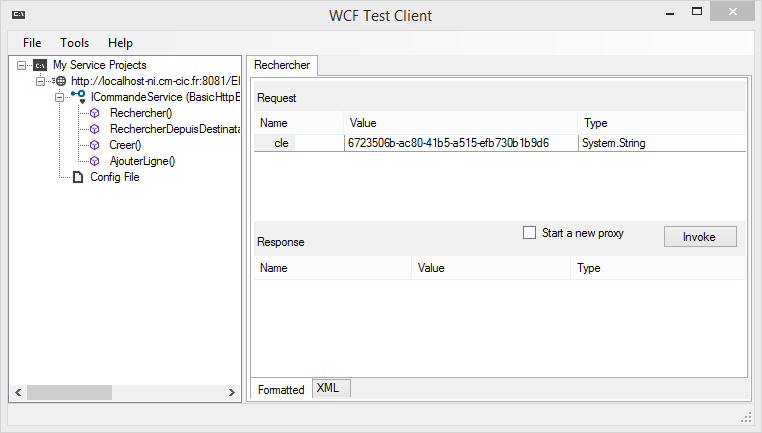


Figure - Formulaire d'appel de *WCF Test Client*

Le service apparaît alors dans le volet de gauche. La liste des différentes opérations est disponible sous le nom du service. Double-cliquez sur une opération pour faire apparaître un formulaire d’appel dans le volet de droite. La partie supérieure vous permet de renseigner les différents paramètres de l’appel et la partie inférieure contient la réponse.

**Attention !** Tous les types de paramètre ne sont pas supportés par *WCF Test Client*. Si vos paramètres sont complexes, vous devrez écrire un client en C# pour appeler votre service.

# Développement sur une application batch

Cette section ne s’adresse qu’aux développeurs qui seront amenés à développer des applications de type batch.

## Bases de données

Contrairement au moteur qui embarque son propre client SQL, les batchs accèdent aux bases de données en utilisant un client Oracle installé séparément et des sources de données ODBC configurées sur la machine.

Avant de développer un batch, vous devez :

1. Demander l’installation du driver Oracle 11g sur votre machine
2. Demander l’installation du fichier TNS des bases de données
3. Configurer, ou demande à configurer les sources de données utilisateur ODBC suivantes

|  |  |  |  |
| --- | --- | --- | --- |
| Base | Data Source Name | TNS Service Name | User ID |
| CRM (développement) | EIF\_CRM\_DEV | EIFCRM\_TEST | EIF |
| Usages (développement) | EIF\_USAGES\_DEV | EIFUSAGES\_TEST | EIFF |

## Dépendances techniques

Les applications batch dépendent des librairies techniques du paquet EIFA\_BCH\_SYSTEME. Ce paquet doit être installé sur votre poste de développement avec PACMAN.

Le projet qui contient le batch doit référencer les assemblages suivants :

* C:\appli\statro\EIFA\_BCH\_SYSTEME\EIT.Fixe.Batch.Data.dll
* C:\appli\statro\EIFA\_BCH\_SYSTEME\EIT.Fixe.Batch.Systeme.dll
* C:\appli\statro\EIFA\_BCH\_SYSTEME\EIT.Fixe.Batch.Traces.dll

## Lancer un batch sur son poste de développement

La classe du batch est une classe **publique** qui doit hériter de **EIT.Fixe.Batch.Systeme.Traitement**. La méthode **Executer()** de **EIT.Fixe.Batch.Systeme.Traitement** doit être surchargée et contenir le code du batch.

Le projet qui contient la classe du batch doit être le projet de démarrage de la solution. Il devrait apparaître en gras dans la liste des projets de la solution. Si ce n’est pas le cas, faites un clic droit sur le projet puis sur ***Set as StartUp Project*** dans le menu qui apparaît.

Il doit posséder un fichier de configuration **App.config** qui contient les paramètres suivants :

<appSettings>  
 <add key="CRM.DEV" value="KWbB5Pr2VEGwEt9GO3x36A==|cPZoE4xYrV59WnZ+Mc3ZbgjPtMdHbpELUeQBaGL3xngZaWxZSNPunhqOpaT6Yxxn" />  
 <add key="USAGES.DEV" value="g5tz+m/pSRpsJV+d7I5oog==|7ivIHuSDadKJCKHPhdLBTsxLmZjFogNsekX3gTM44uG+ADvYLHwg6Bty6ibsp4RG" />  
 <add key="Version" value="88"/>  
</appSettings>

Un programme externe doit être utilisé pour le démarrage du batch. Accédez aux propriétés du projet, puis dans la section ***Debug***, cochez l’option ***Start external program*** et renseignez l’exécutable suivant : C:\appli\statro\EIFA\_BCH\_SYSTEME\EIT.Fixe.Batch.Lanceur.exe. Les arguments de la ligne de commande doivent être de la forme "NomCompletDeLaClasseDuBatch, NomDeL’Asseblage", par exemple "EIT.Fixe.MonBatch.MaClasseDeBatch, EIT.Fixe.MonBatch".

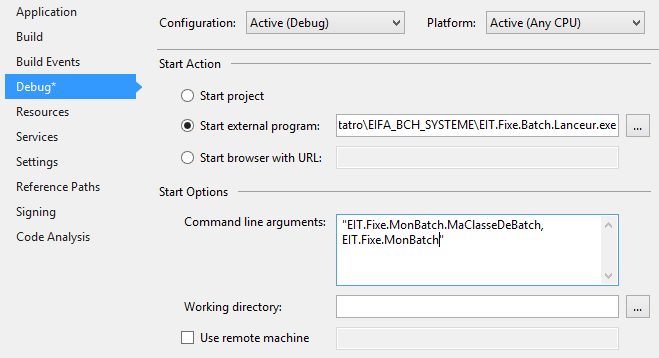


Figure 19 - Configuration des propriétés de lancement d'un batch en debug

Une fois cette configuration effectuée et enregistrée, le batch peut être lancé en mode debug.

# Liens utiles

Cette section contient une liste de liens vers les outils de développement.

Team Foundation Servier : <http://tfs.eit.cm-cic.fr:8080/tfs/EITCollection/Fixe>/

SonarQube : <http://sonar.eit.cm-cic.fr:9000/>

Web Services, serveur d’intégration continue : <http://xcmci53l.formation.cm-cic.fr/EIT.Fixe>/

Web Services, serveur de l’environnement NI : <http://nrjmobile-ws-dev.cm-cic.fr/EIT.Fixe/>