

# Ma première application Windows Azure

La maturité de Windows Azure n'est plus à démontrer. Il en est de même pour son environnement de développement. Pour vous permettre d'en découvrir la puissance et d'en apprécier sa simplicité, nous allons vous guider au travers des différentes étapes pour créer, puis configurer un projet de type Cloud sous Visual Studio et effectuer un premier déploiement sur Windows Azure.

**A**vant de commencer, allons faire un petit tour du côté des habituels prérequis. La première étape consiste donc à vérifier et éventuellement installer et configurer votre socle applicatif. Vous aurez besoin :

- Du Framework .Net, en version 4.0, avec le SP1 installé
- D'une version de Visual Studio 2010, avec le SP1 installé
- Du Windows Azure SDK, le kit de développement complémentaire
- Une version de SQL Server 2008 R2 ou SQL Server 2008 R2 Express, afin de simuler SQL Azure si vous envisagez d'utiliser une base de données.

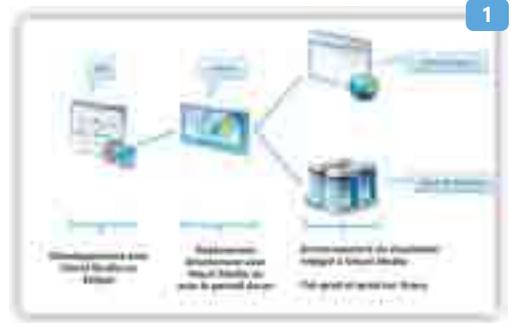
Vous pouvez télécharger le SDK ici : <http://msdn.microsoft.com/fr-fr/windowsazure>, son installation ne pose aucune difficulté. Voici le modèle de développement et de déploiement avec Windows Azure (**Fig. 1**).

Le SDK 1.6 intègre :

- Des modèles de programmation Windows Azure
- Les API pour lancer les machines virtuelles
- Un environnement de simulation Azure en local
- Les outils de packaging et de déploiement automatique
- Des exemples d'applications et un kit de formation disponible en complément dans le Windows Azure Platform Training Kit (<http://www.microsoft.com/download/en/details.aspx?id=8396>)

## Création du Projet sous Visual Studio

La création du projet est très simple : lancez Visual Studio en mode administrateur. C'est important pour pouvoir ultérieurement lancer l'émulateur. Une fois dans Visual Studio, sur la page d'accueil cliquez sur « New Project » ou allez dans « File » → « New » → « Project ». Donnez ensuite un nom à votre projet (**Fig. 2**). L'étape suivante consiste à ajouter des rôles à votre projet. Windows Azure dispose de machines virtuelles appelées « rôles » pour exécuter les



composants applicatifs. La notion de rôle est nouvelle et propre aux projets de type Windows Azure. Pour tirer parti des bénéfices d'Azure en termes de disponibilité et de montée en charge, une application doit suivre les règles suivantes :

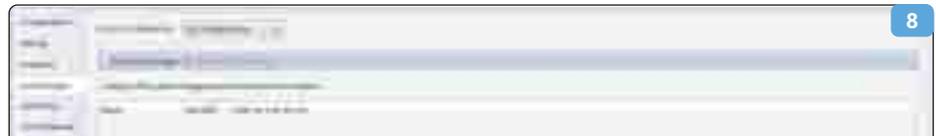
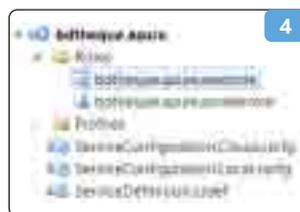
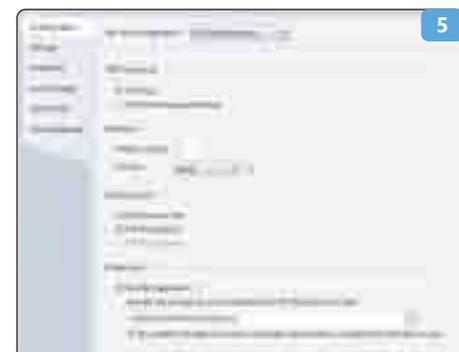
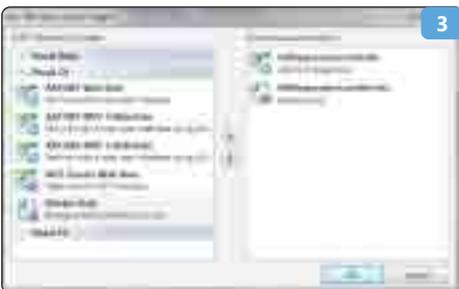
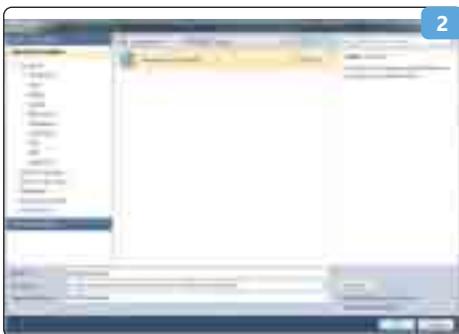
- Une application Windows Azure est construite sur un ou plusieurs rôles
- Chaque rôle doit être déployé au minimum sur deux instances
- Une application Azure doit se comporter correctement si une instance tombe. Il est important de conserver un mode dit « stateless » : les applications Windows Azure ne doivent pas gérer d'état pour fonctionner de façon optimale. Dans notre exemple, l'application va implémenter deux rôles (**Fig. 3**) :

- Un Web Role Basé sur IIS, qui exécute les applications Web et utilise ASP.NET, WCF, PHP, etc.
- Un Worker Role utilisé pour les traitements en arrière-plan

Une fois ces deux étapes passées, le projet est créé, et déjà prêt à être déployé. Un projet Windows Azure contient en fait plusieurs projets accessibles depuis l'explorateur de projets :

- Un projet global décrivant les rôles de l'application et leurs configurations.
- Autant de projets que de rôles créés.

Il est tout à fait possible, grâce à l'émulateur



Windows Azure, de lancer notre application en local et la débbugger en appuyant sur F5. Grâce à celui-ci il est possible d'exécuter notre projet en local et de simuler les stockages Azure, cela permet donc de tester et débbugger le code utilisant celui-ci. Avant de déployer cette première application vide, il est important de s'arrêter sur la configuration de nos différents rôles.

Le Worker Role et le Web Role ont à peu de chose près la même configuration. Pour voir la configuration de notre Web Role, il faut cliquer sur le bouton droit sur l'un de vos deux rôles dans le dossier Roles de notre projet Cloud et sélectionner Properties (Fig. 4).

La configuration se divise en plusieurs parties.

**Configuration (Fig. 5) :** La première concerne la configuration générale du Role. C'est ici que l'on va définir le nombre d'instances que l'on va utiliser. Ce paramètre est très important, car c'est lui qui va contrôler la notion de « Load Balancing » et de haute disponibilité de votre solution. Il est conseillé de paramétrer plusieurs instances et non une seule. Microsoft ne garantit d'ailleurs un SLA élevé que si deux instances minimum sont mises en production. Un autre paramètre important est la taille de la machine virtuelle, configurée selon la taille de l'application. Les informations relatives aux tailles des machines virtuelles sont disponibles ici : <http://www.microsoft.com/france/windows-azure/offres/tarifcation.aspx>.

On peut également choisir de lancer l'application en http ou https (si des certificats SSL sont présents sur le portail Azure). Dans le cas d'utilisation d'https, il sera nécessaire d'installer les certificats sur le serveur.

**Settings (Fig. 6) :** Cette deuxième partie permet de stocker différents paramètres de configuration (chaînes de connexion, emplacement du serveur de stockage Azure, Diagnostics ...).

**Endpoints (Fig. 7) :** Cette partie permet de définir les points de communication (http, https, ports, ...).

**Local Storage (Fig. 8) :** Cette partie permet de déclarer sur le File System local de l'instance en production sur Windows Azure des espaces de stockage. (Pour créer un système de cache par exemple).

**Certificates :** Cette partie permet de rajouter des certificats SSL, qu'il faut avoir installés auparavant sur le serveur via le portail.

**Virtual Network :** Cette dernière partie permet de gérer l'utilisation d'Azure Connect, qui autorise votre application à se connecter sur un serveur interne à l'entreprise par TCP. Voilà, vous avez maintenant un projet, vide, mais correctement configuré pour pouvoir effectuer votre premier déploiement sur Windows Azure.

## DEPLOIEMENT SUR WINDOWS AZURE

Il peut être effectué de deux manières différentes :

- Dans le portail Windows Azure
- Dans Visual Studio 2010 : le processus de déploiement est complètement intégré à l'outil de développement. Il peut aussi être pris en charge dans le processus d'intégration de Team Foundation Server.

Alors prenons le temps de tester les deux options.

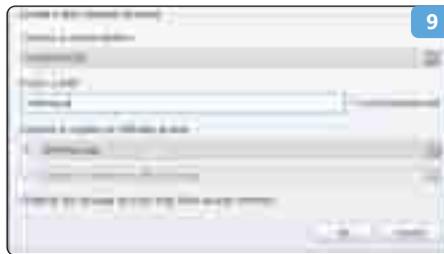
## DEPLOIEMENT VIA LE PORTAIL WINDOWS AZURE

Connectez-vous avec votre profil sur le portail Azure. Sur la partie gauche de l'écran sont proposés dans un menu les différents services et actions possible. Pour créer le 1er service, il faut cliquer tout simplement sur l'icône « New Hosted Service ». Qui donne accès à l'écran de configuration. Donnez un nom pour votre service, qui sera donc accessible via l'URL nom du service. cloudapp.net. Pour ce premier déploiement c'est l'option « stage environment » qui est sélectionnée ; il s'agit bien de tester la solution, pas de la mettre en production. La solution sera déployée sur les datacenters Européens de la plateforme Windows Azure.

Ce point étant traité, il faut maintenant s'intéresser à la partie stockage des données : Après avoir cliqué sur « Storage Accounts », puis l'icône « New Storage Account » (Fig. 9).

Une fois cette étape franchie, nous avons donc deux comptes actifs : un pouvant héberger sur plusieurs machines virtuelles nos services applicatifs, et un second pouvant héberger nos machines virtuelles dédiées aux données.

Vous pouvez profiter de cette étape pour créer si vous le souhaitez un serveur SQL Azure et votre première base de données. Les étapes de création et de configuration des ressources Azure sont maintenant terminées, nous allons pouvoir tester la mise en ligne de la solution « vide » : Il suffit d'effectuer un clic droit sur votre projet dans Visual Studio et de sélectionner Package (Fig. 10). Les deux fichiers demandés



par le portail sont générés automatiquement. Il ne reste plus qu'à copier le chemin des fichiers, les coller sur le portail et valider la création du service hébergé. Au bout de 10 à 15 minutes maximum, votre application sera disponible sur le cloud via l'URL de développement ou de production (selon votre choix précédemment effectué) (Fig. 11). Votre première application Windows Azure est en ligne, félicitations !

## DEPLOIEMENT VIA VISUAL STUDIO

Il existe une deuxième méthode pour déployer une application sur Windows Azure. Via Visual Studio : au lieu de régénérer les packages de l'application et de les uploader sur le portail, quelques manipulations simples vont vous le permettre depuis votre Visual Studio. Pour cela, sélectionnez le projet racine, puis cliquez sur le bouton droit pour sélectionner « Publish ».

Pour le premier déploiement, une configuration initiale doit être effectuée. Il faut en effet que l'on s'authentifie sur le portail via Visual Studio. Ceci se fait via une authentification par certificats, ce qui est bien compréhensible. Ensuite, les écrans de publication s'enchaînent, un premier permettant de sélectionner son abonnement Windows Azure, suivi des paramètres de mise en staging de l'application, avant d'obtenir votre accord final (Fig. 12).

Voilà, vous avez en main les principales étapes pour créer votre première application Windows Azure et la mettre en production. A vous de jouer maintenant !

François Mérand  
National Practice Leader – Alliance Microsoft  
SOGETI France  
<http://www.dotnetrangers.net>

Jason De Oliveira  
Practice Manager & Solutions Architect | MVP  
C# - SOGETI France  
Blog: <http://jasondeoliveira.com>

