

Gestion IIS 7 en C#

par [David Tavan](#)

Date de publication : 6 mars 2007

Dernière mise à jour : 10 mars 2007

Voici une introduction à la gestion de IIS 7 grâce à Visual Studio 2005 et dans notre cas en C#. Tout d'abord il faut bien préciser que je parle uniquement de IIS 7, de fait cet article concernera que les OS suivants : VISTA et " longhorn server ".

- I - Création d'un site web
- II - Création d'un pool d'application
- III - Assignation d'un pool existant à un site.
- IV - Création d'une application et d'un répertoire virtuel
- V - Lister les sites présents dans IIS 7

I - Création d'un site web

Pour commencer nous allons voir comment créer un site web. Pour ce faire nous ajouterons à nos projets une référence faire Microsoft.Web.Administration. Pour créer un site nous devons fournir les informations suivantes à notre application :

- a Un nom pour le site
- b Un port
- c Un chemin physique

Nous déclarerons une instance de `ServerManager` et une instance de `Site`, ce qui nous donne une fonction du type :

```
C#
static void createSite(string strPath, string strName, int iPort)
{
    _srvMgr = new ServerManager();
    _site = _srvMgr.Sites.Add(strName, strPath, iPort);
    _site.ServerAutoStart = true;
    _srvMgr.CommitChanges();
}
```

L'utilisation devient alors simpliste, par exemple je veux créer le site test pointant sur le dossier d:\test avec le port 8080 :

```
C#
createSite(@"d:\test\", "test", 8080);
```

Comme prévu le site est créé pour vérifier allons faire un petit tour dans le gestionnaire de IIS :

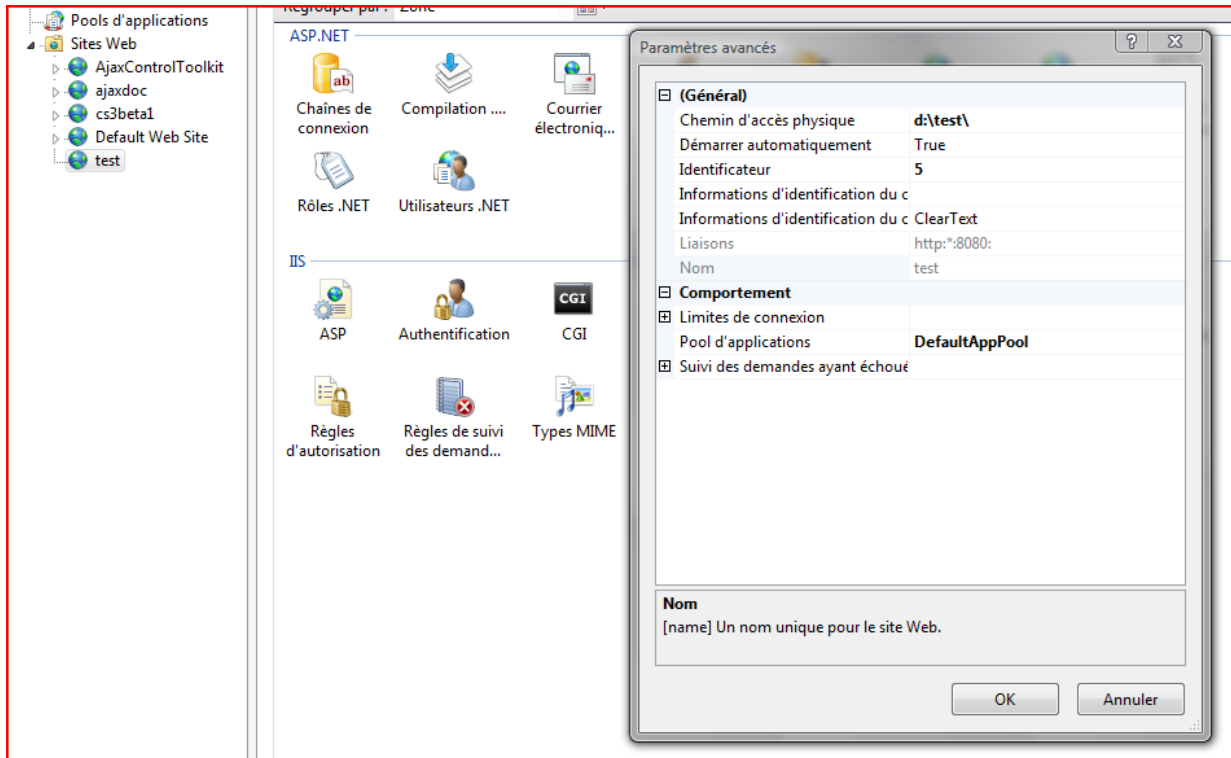


fig 1

Comme prévu le site est créé.

II - Création d'un pool d'application

Nous allons maintenant changer le pool d'application pour notre site,

tout d'abord créons un nouveau pool d'application :

Pour ce faire nous allons créer une fonction et instancier une ajouter une instance de ApplicationPool.

Au niveau des déclarations :

```
C#
static ServerManager _srvMgr;
static Site _site;
    static ApplicationPool appPool;
```

Maintenant la fonction permettant de créer un nouveau pool et de l'assigner à un site :

```
C#
static void createAndAssignNewPool(string strSite, string strPoolName)
{
    _srvMgr = new ServerManager();

    _srvMgr.ApplicationPools.Add(strPoolName);
    _srvMgr.Sites[strSite].Applications[0].ApplicationPoolName = strPoolName;

    appPool = _srvMgr.ApplicationPools[strPoolName];
    appPool.ManagedPipelineMode = ManagedPipelineMode.Classic;
    _srvMgr.CommitChanges();
    appPool.Recycle();
}
```

L'utilisation de cette fonction sur notre site précédemment créé ce fait de la sorte :

```
C#
createAndAssignNewPool("test", "newPoolTest");
```

Histoire d'en avoir le c#ur net regardons dans le gestionnaire IIS :

Le pool est-il créé ?

| | | | | | |
|----------------------|---------|------|-----------|----------------|---|
| Classic .NET AppPool | Démarré | v2.0 | Classique | NetworkService | 3 |
| DefaultAppPool | Démarré | v2.0 | Intégré | NetworkService | 1 |
| newPoolTest | Démarré | v2.0 | Classique | NetworkService | 1 |
| ReportServer | Démarré | v2.0 | Classique | NetworkService | 2 |

fig 2.

La réponse est oui la seconde question suivante est de savoir si le site test à bien pris en compte le nouveau pool :

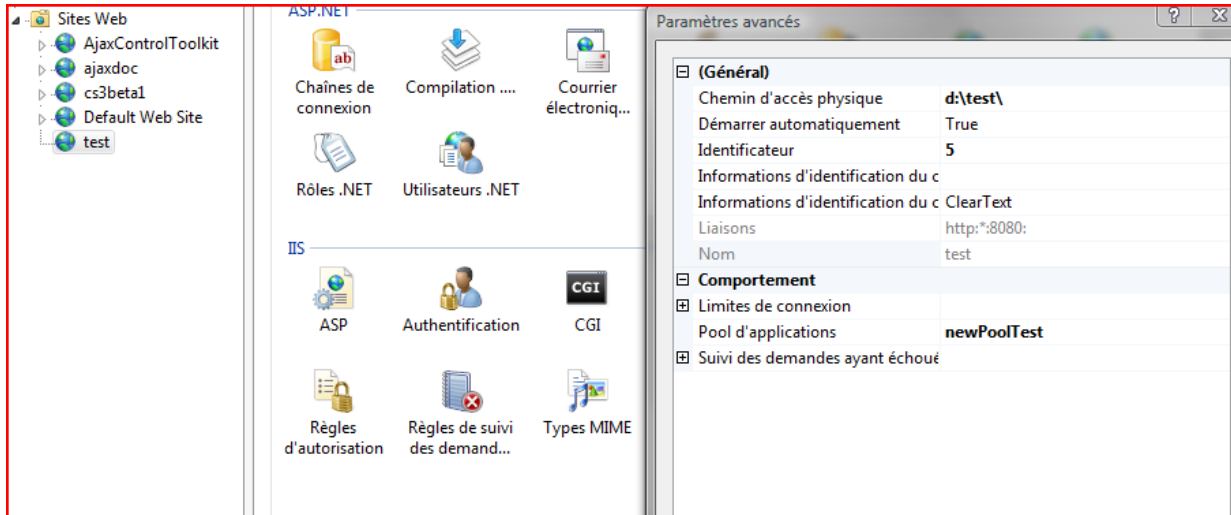


fig 3

Bon tout est comme prévu. (Heureusement :)

III - Assignation d'un pool existant à un site.

Pour commencer nous devons récupérer les pools existant,

pour ce faire j'utilise comme je l'ai écrit ici : <http://www.developpez.net/forums/showthread.php?t=283047> ,

c'est-à-dire je crée deux fonctions :

Fonction permettant de récupérer les pools existants

```

static IEnumerator GetExistingPools()
{
    _srvMgr = new ServerManager();
    IEnumerator enu =
        _srvMgr.ApplicationPools.GetEnumerator();
    return enu;
}

static List<string> existingPools()
{
    IEnumerator enumPools = GetExistingPools();
    List<string> lstPools = new List<string>();

    int i = 0;

    while (enumPools.MoveNext())
    {
        lstPools.Add(_srvMgr.ApplicationPools[i].Name);
        i++;
    }
    return lstPools;
}

```

Comme c'est une application console je vais faire afficher la liste des pools disponibles :

Affichage des pools existants

```

static void listExistingPool()
{
    List<String> lstPools = new List<string>();
    lstPools = existingPools();

    foreach (String p in lstPools)
    {
        Console.WriteLine(p.ToString());
    }
}

```

Maintenant dans le Main nous allons afficher les pools et demander à l'utilisateur de choisir un pool pour un site :

Affichage et renseignement d'un pool

```

listExistingPool();
string strPool;
Console.WriteLine("Choisir un pool");
strPool = Console.ReadLine();
assignPool("test", strPool);

```

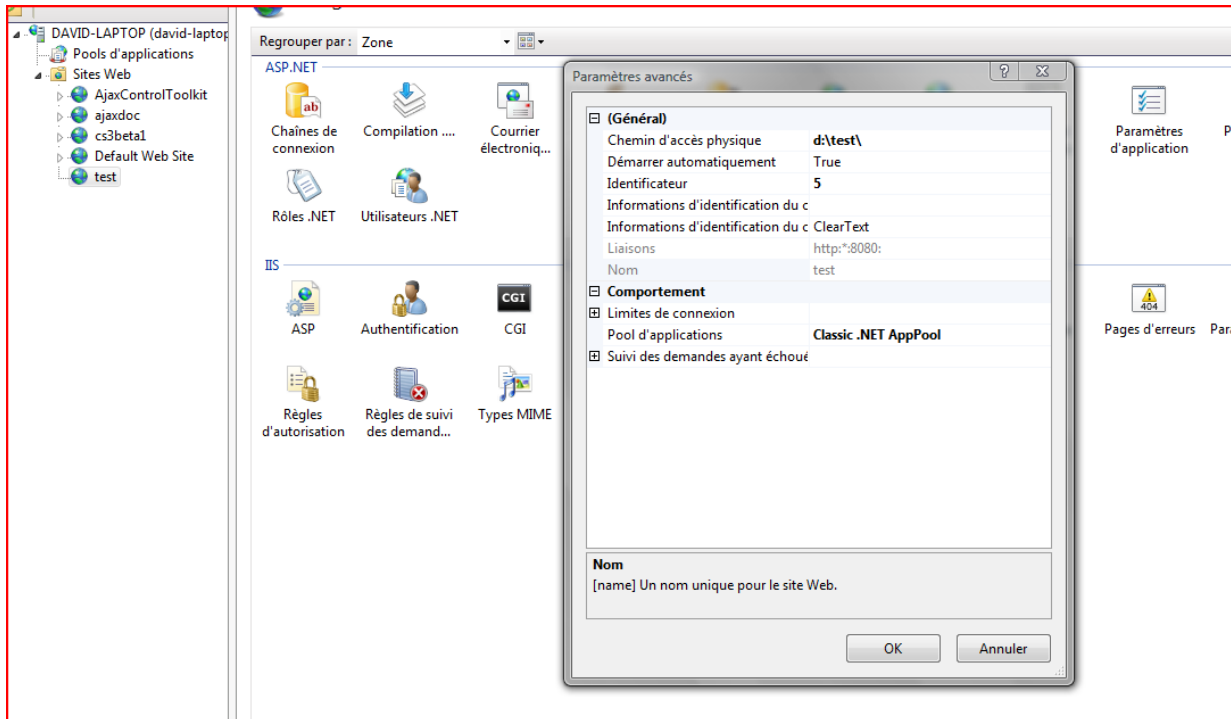
A l'exécution cela nous donne :

```
DefaultAppPool
Classic .NET AppPool
ReportServer
newPoolTest
Choisir un pool
```

Il suffit maintenant de taper le nom du pool souhaité puis entrée :

```
DefaultAppPool
Classic .NET AppPool
ReportServer
newPoolTest
Choisir un pool
Classic .NET AppPool
ENTER to END
tati
tati
tati
tati
```

Un petit tour dans IIS :



Le pool a bien changé :)

IV - Création d'une application et d'un répertoire virtuel

TODO : je vais ajouter la création d'application et de répertoires virtuels

V - Lister les sites présents dans IIS 7

à faire