

## Quel avenir pour **PHP** ?

- L'échec de PHP 6
- L'hypothétique roadmap
- Quels outils de développement ?
- Forces et faiblesses du langage

Cloud computing

## **Windows Azure :** le développeur dans les nuages

- Migration, développement et déploiement dans le cloud
- Gérer les données dans Azure
- L'architecture du cloud

### **Mobile**

Développer pour les nouveaux Palm

### **Google**

Maîtriser Google Analytics

### **C #**

Au cœur de DirectX 11

### **Embarqué**

Robotics Studio : programmer son robot

### **CARRIERES**

Les salaires de la crise

### **WEBMASTER**

La vidéo sur le web

### **Sharepoint 2010**

Les nouveautés pour le développeur

### **GreenIT**

Intel Energy Checker : traquer le profil énergétique d'une application

© istockphoto.com/Evirgen

M 04319 - 131 - F: 5,95 €



Printed in France - Imprimé en France - BELGIQUE 6,45 €  
SUISSE 12 FS - LUXEMBOURG 6,45 € - DOM Surf 6,90 €  
Canada 8,95 \$ CAN - TOM 940 XPF - MAROC 50 DH



# DÉVELOPPEZ 10 FOIS PLUS VITE

## WINDEV®

**ENVIRONNEMENT  
PROFESSIONNEL  
DE DÉVELOPPEMENT  
(AGL)**

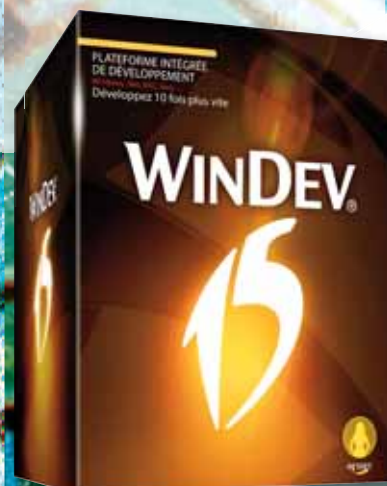
Créez des applications pour Windows, Internet, Mobile et SaaS  
Java, .Net, PHP, J2EE, Webservice,  
XML, Ajax, Linux, Android,  
Client riche, Base de Données...

**AGL N°1 EN FRANCE**

Demandez le dossier + DVD gratuit

**Tout est inclus**  
Gère le cycle  
**complet** de  
développement.  
**Support  
Technique  
gratuit**

**555  
FOUVERÉTÉS**



**www.pcsoft.fr**



Elu «Langage le  
plus productif du  
marché»

**VERSION  
EXPRESS  
GRATUITE**  
Téléchargez-la !

Fournisseur Officiel de la Préparation Olympique



► Dossier gratuit 200 pages sur simple demande. Tél: 04.67.032.032 info@pcsoft.fr



# sommaire

## \\ actus

En bref .....	6
Agenda .....	54

## \\ webmaster

Maîtriser la diffusion de vidéo au format Flash .....	12
Smooth Streaming : un streaming vidéo efficace .....	15

## \\ gros plan

### SharePoint 2010 :

Une version au top pour le développeur .....	18
--	----

## \\ outils

Intel Energy Checker : votre logiciel est-il vert ? .....	24
---	----

## \\ dossier

### PHP a-t-il encore un avenir ?

La communauté PHP : son organisation .....	30
PHP : choisir un bon outil .....	32
Un éléphant aux pieds d'argile ? .....	34
Unicode et les conséquences sur PHP 6 .....	36
PHP a encore un bel avenir devant lui .....	40

## \\ dossier

### Windows Azure : le cloud computing pour les développeurs

L'architecture de Windows Azure .....	46
Le prix de l'Azure .....	48
Créer son premier service Windows Azure .....	49
Le stockage dans le nuage Azure .....	51
Intégrer les applications et exposer des services grâce à AppFabric .....	55
Exemple d'une migration ASP.net vers Azure .....	59

## \\ technique

Choisir sa forge pour ses projets libres .....	62
--	----

## \\ carrière

Entre crise et restructuration, un marché qui se redresse .....	64
---	----

## \\ code

Microsoft Robotics Developer Studio : la robotique étend ses frontières (2e partie) .....	66
Partez à la découverte de DirectX 11 .....	70
Interroger Google Analytics .....	74
Ecrire des applications pour Palm WebOS .....	78

## \\ temps libre

Les livres du mois .....	82
--------------------------	----



6



24



28



44



78



66

L'info continue sur [www.programmez.com](http://www.programmez.com)

**CODE**

Les sources  
des articles

**NOUVEAU**

Livres blancs :  
langages, outils...

**TÉLÉCHARGEMENT**

Les dernières versions de vos  
outils préférés + les mises à jour

**QUOTIDIEN**

Actualité, Forum  
Tutoriels, etc.

# MODEL2CODE

Agile Development in action!

CERTAINS AGL VOUS PROMETTENT DE DÉVELOPPER  
10 FOIS PLUS VITE?

**ET SI VOUS VOUS LAISSIEZ... DIRIGER PAR LES MODÈLES ?  
NE DÉVELOPPEZ PLUS, MODÉLISEZ !**

Passez à la vitesse du Model Driven!

Maquette vos IHMs en HTML ou MXML et modélisez simplement vos processus métier sous forme de diagrammes UML, nos générateurs les transforment instantanément en application Spring, Java EE, Flex ou Improve Foundations, prête à être déployée !

Nouveau ! Plugin CRUD Booster

Créez en quelques minutes vos prototypes et accélérez considérablement vos projets applicatifs. Générez une application CRUD fonctionnelle (y compris les IHMs) à partir d'un simple diagramme de classe !

**VERSIONS D'ÉVALUATION GRATUITES  
& TUTORIAUX :**

**WWW.MODEL2CODE.COM**

Model2Code propose des ateliers agiles et collaboratifs de génération d'applications web et riches, combinant les meilleures technologies Model Driven du marché : l'outil de modélisation UML Magicdraw® associé au moteur de génération par transformation de modèles BLU AGE®.



 magicdraw™

 BLU AGE  
SOLE MODEL TRANSFORMATION

© 2004 Altkomputer GmbH & Partner AG





## 15-0, 0-15 ou 15-A

De temps en temps, on a une impression d'être sur un radeau, la mer est calme, sans vagues, puis tout à coup, une tempête se lève et manque de nous emporter dans les abysses. Et là en quelques jours, ce fut la tornade : Mandriva menacé de disparaître, Apple qui attaque une nouvelle fois Flash, Adobe qui pactise avec Android mais continue de faire les yeux doux à iPhone et autres iPad, Google qui lance la guerre de la vidéo sur le web et de HTML 5, VMware qui met du Java partout sur le cloud, Sybase qui se fait racheter et Palm qui suit le même chemin... Bref, que d'actions !

Il y a quelque chose de paradoxal chez Adobe. Il faut avouer que leur position est loin d'être confortable. Apple qui fustige Flash pour x raisons (et parfois avec justesse il faut le dire) et pousse HTML 5 tandis qu'Adobe continue à travailler sur iPad et iPhone pour Air 2. Mais surtout, Adobe n'hésite pas à aller voir Mozilla et Google pour apporter le support de Flash sur Android et Chrome, tout en voulant briser la folie des plug-ins avec deux éditeurs de navigateurs, qui eux-mêmes poussent fortement HTML 5 et des formats vidéo concurrents de Flash... D'ailleurs, Google, lors de la conférence I/O a annoncé la mise en open source du fameux codec VP8 et pousse de plus en plus fortement HTML 5 pendant qu'Adobe démontre le support de Flash chez Google et dévoile un module HTML 5 pour Dreamweaver. De quoi vous donner le tournis non ? Mais on n'arrête pas la bizarrerie car pour faire tourner la prévisualisation, le module Dreamweaver s'appuie sur WebKit... En poussant l'analyse, on constate qu'une guerre souterraine fait rage sur les moteurs de rendu. C'est un peu WebKit contre Gecko. Pour caricaturer : Apple vs Mozilla. Reconnaissons que WebKit a su imposer à de nombreux éditeurs son moteur : Nokia, Google, Adobe, Apple. Gecko doit impérativement reconquérir pour ne pas perdre de précieux points. Car, Firefox a beau avoir une belle communauté d'utilisateurs, rien n'est immuable, surtout dans le monde impitoyable du web.

En réalité, Adobe tente de préserver son marché (Flash) tout en gardant un œil sur les concurrents et même en flirtant un peu avec (HTML 5). Adobe ne pouvait pas rester à l'écart de HTML 5 car même si les spécifications ne sont pas figées et officielles avant fin 2010, voire début 2011, les grandes manœuvres ont déjà débuté. Et le risque de marginaliser Flash à terme est loin d'être négligeable. C'est le cœur métier d'Adobe qu'il faut défendre. Et que l'on soit d'accord ou non avec Steve Jobs, a-t-il réellement tort de vouloir privilégier des applications réalisées et optimisées pour ses terminaux mobiles ? Vous verrez que la réponse est bien loin d'être aussi simple qu'il n'y paraît.

La partie vidéo concerne la prochaine guerre ouverte et là aussi, Adobe doit absolument assurer ses arrières. Google a placé une jolie bombe à retardement en mettant en open source VP8 avec le projet WebM. Concurrent direct de H.264, Google met en difficulté Microsoft et Apple mais aussi Adobe avec FLV. Si Flash perd cette guerre, les conséquences seront considérables car de facto, on ne passera plus par le Flash Player, mais directement par le navigateur... On comprend alors l'intérêt d'Adobe de travailler avec Google et Mozilla sur une nouvelle génération de plug-in tout en voulant capter de nouveaux marchés avec l'initiative Open Screen. Oui, vous aurez accès au code source du player Flash, oui vous aurez des facilités et une licence gratuite, mais vous devrez installer le player sur les terminaux et matériels capables de le supporter. Une manière efficace pour rendre incontournable une technique.

Heureusement que le tennis est plus simple à suivre que l'informatique !

■ **François Tonic**  
Rédacteur en chef  
ftonic@programmez.com

**Éditeur :** Go-02 sarl, 21 rue de Fécamp 75012 Paris - diff@programmez.com.

**Rédaction :** redaction@programmez.com

**Directeur de la Rédaction :** Jean Kaminsky.

**Rédacteur en Chef :** François Tonic - ftonic@programmez.com. **Ont collaboré à ce numéro :** F. Mazue, G. Delamarre, P. Davy. **Experts :** L. Lemoine, J.-C. Vasselon, J. Dollon, J. Tayeb, F. Hardy, C. P. de Geyer, N. Silberman, C. Villeneuve, C. Cabesse, B. Chauvin, T. Laurens, D. Puig, B. Lefrançois, N. Clerc, P. Lamarche, A. Collins, C. Siffroi, C. Ouaknine, P. Cauchois, D. Catuhe, Y. Bell.

**Illustration couverture :** © istockphoto.com/Evirgen

**Publicité :** Régie publicitaire, K-Now sarl. Pour la publicité uniquement : Tél. : 01 41 77 16 03 - diff@programmez.com.

**Dépôt légal :** à parution - Commission paritaire : 0712K78366 ISSN : 1627-0908. Imprimeur : S.A. Corelio Nevada Printing, 30 allée de la recherche, 1070 Bruxelles Belgique. **Directeur de la publication :** J-C Vaudecrane

**Abonnement :** Programmez 22, rue René Boulanger, 75472 Paris Cedex 10  
Tél. : 01 55 56 70 55  
abonnements.programmez@groupe-gli.com  
Fax : 01 40 03 97 79 - du lundi au jeudi de 9h30 à 12h30 et de 13h30 à 17h00, le vendredi de 9h00 à 12h00 et de 14h00 à 16h30. **Tarifs**  
abonnement (magazine seul) : 1 an - 11 numéros France métropolitaine : 49 € - Etudiant : 39 € - CEE et Suisse : 55,82 € - Algérie, Maroc, Tunisie : 59,89 € - Canada : 68,36 € - Tom : 83,65 € - Dom : 66,82 € - Autres pays : nous consulter. **PDF :** 30 € (Monde Entier) souscription exclusivement sur [www.programmez.com](http://www.programmez.com)

L'INFO  
PERMANENTE  
[WWW.PROGRAMMEZ.COM](http://WWW.PROGRAMMEZ.COM)



## PROCHAIN NUMÉRO

N°132 juillet/août 2010  
parution 30 juin

### ✓ Dossier Smartphone

- Découvrez Windows Phone 7
- Développez pour iPhone et Android

**Votre premier projet iPad**

### ✓ Eclipse e4

Le futur d'Eclipse se dévoile



■ Bien que **Mozilla** ne veuille pas (pour le moment ?) du format vidéo H.264 dans Firefox et qu'il lui préfère Ogg Theora, un fork vient de se lancer pour inclure H.264 dans le renard.

Le projet se nomme wildfox et apportera le support de la balise vidéo de HTML5. La guerre de la vidéo sur le web ne fait que commencer !

site : <http://wildfox.sourceforge.net/>

■ **Samsung** vient de lancer un SDK pour son propre système : BADA. Si au départ le SDK pour BADA était réservé à ses seuls partenaires, la firme coréenne a pris la décision de le rendre public. Pour y accéder, il suffit de s'inscrire sur le site Samsung Bada Developer. Le SDK contient un IDE mais n'est disponible que sous Windows. Pour faire bonne mesure, Samsung organise un concours pour les développeurs BADA.

Le premier prix est conséquent : 300 000 dollars.

■ **Google** continue à poser les jalons de Chrome et surtout de ChromeOS. L'éditeur sort un SDK pour le Native Client. L'objectif ici est de pouvoir exécuter du code natif dans une application web, un navigateur, voire un système dans le cloud comme ChromeOS. Le projet est très loin d'être finalisé et Google attend le retour des développeurs. Le SDK fonctionne sur Windows, MacOS X et Linux en 32 bits pour le moment (mode x86).

■ Quel avenir pour **Mandriva** ? Depuis plusieurs mois la question se pose et la situation semblerait s'aggraver rapidement. Dernièrement, Linagora s'est dit intéressé par un rachat partiel de l'éditeur français. LightApps, société anglaise, est aussi en négociation. Mandriva va-t-il disparaître du paysage Linux ? Ce serait une victime non négligeable...

## Salons

# Datacenter et cloud computing à l'honneur à Paris

**L**es 5 et 6 mai derniers, s'est tenu le salon Cloud et Data center. Quel bilan en tirer ? Premier constat, une présence finalement discrète du cloud computing de type Amazon EC2, Ubuntu, Azure, etc. On a surtout pu voir du cloud privé et beaucoup de virtualisation, ce qui nous a un peu surpris. Avec bien entendu, une belle partie matérielle, à cause de la déclinaison data center.

Il manquait sans doute une zone interopérabilité ou même Ask to me pour des experts. Mais au-delà de ces manques, notons une bonne fréquentation et les éditeurs non présents comme

Microsoft, IBM, le seront peut-être l'an prochain.

L'autre information importante était le lancement par OW2 de Cloudware Initiative, nom de code JetStream. Cela permettra une interopérabilité dans le cloud, d'avoir des solutions open source pour la gestion des ressources, des connecteurs, des connexions entre les acteurs et les communautés. L'open source pour OW2 doit jouer un rôle dans le cloud et il entend bien y apporter sa contribution. L'ouverture est indispensable. Il ne devrait y avoir de barrières ni à l'entrée, ni à la sortie et pas de discrimination. Il est vital d'assu-

rer une interopérabilité, d'utiliser des licences libres, d'être neutre technologiquement et transparent. Il existe déjà des outils de gestion pour la virtualisation, le cloud comme OpenNebula, Apache Hadoop et pour la partie infrastructure comme Xen, KVM. Le projet JetStream doit pouvoir ouvrir une pile cloud intégrée et ouverte, être une ligne directrice pour les outils et initiatives cloud ouvert, avec un réseau de partenaires. OW2 présente comme des standards les initiatives autour de DTMF, OCCI, CCIF, Cloud Security Alliance, or, ce n'est pas le cas. Rien n'est clairement défini.

## Concours

# ImagineCup 2010 : objectif Pologne pour nos finalistes !

**L'**édition 2010 du grand concours étudiant mondial de Microsoft, qui aura lieu en Pologne début juillet, rentre dans sa dernière ligne droite. Mardi 4 mai, les finales françaises ont désigné les deux représentants nationaux dans la catégorie développement embarqué et conception logicielle. Plusieurs dizaines d'équipes se sont une fois de plus mobilisées dans toute la France depuis plusieurs mois pour espérer être présentes à Paris et disputer la dernière épreuve : passer devant le grand jury. Dans la catégorie conception logicielle, c'est le projet Babie Angel qui a décroché sa place pour la Pologne. Ce projet a pour ambition de prévenir la mort subite du nourrisson qui tue chaque année des bébés. Une belle initiative. Une mention particulière pour l'équipe de Health Up et une application mobile pour la nutrition et la lutte contre l'obésité. Dans la catégorie développement embarqué, c'est l'équipe GERAS qui a pris la première place. GERAS est une solution comportant de multiples détecteurs installés dans la maison des personnes âgées. Exemple : un tapis intelligent qui



détecte les chutes et qui prévient automatiquement, soit un numéro pré-enregistré et déterminé par le client, soit les urgences. La personne âgée peut également, via un intercom, faire appel aux secours elle-même. Mention spéciale à l'équipe Emanon pour son application de suivi du patient depuis le lieu de l'accident jusqu'au centre de soin. C'était la 2e année que l'équipe participait à ImagineCup avec un bon projet. Espérons que cela permettra de faire décoller les carrières des développeurs de l'équipe !



/\*PARTOUT AUTOUR DE VOUS\*/

Le code. Il a permis de créer toutes ces choses qui nous entourent. Partout où se pose notre regard, il est présent. Tout comme les innombrables opportunités que vous voyez en tant que développeur. Alors, imaginez une nouvelle expérience de travail avec Visual Studio 2010, du design au développement puis au déploiement.

**VOYEZ LA VIE EN CODE.**

ET VOUS, QUE FEREZ-VOUS AVEC VISUAL STUDIO 2010 ?

Rendez-vous sur **VisualStudio.fr**



# Google I/O 2010 : HTML 5 et vidéo open source



L'importante conférence Google I/O qui avait lieu les 19 et 20 mai derniers à San Francisco, a démarré par une session plénière. Selon Google, le web est sans conteste la plateforme du futur. HTML5 est pour Google sur une pente ascendante et irrésistible, et de rappeler au passage qu'Apple soutient HTML5, idem pour Microsoft. Il y a également de nombreux challenges pour le web : déploiement, performances, les applications, l'entreprise. Mais rien ne semble arrêter la croissance du web dans le quotidien.



**L**a keynote a débuté avec un important focus sur HTML 5, preuve que Google y croit beaucoup. On passe de plus en plus de temps sur Internet, il faut donc du contenu. Si jusqu'à présent, les applications les plus populaires étaient sur le desktop, 2005 a marqué une rupture avec l'arrivée en force des applications en ligne, notamment due à Ajax : facebook, youtube, etc. Il faut donc que l'application web en fasse toujours plus : caméra, microphone, graphisme soigné, intégration avec les ressources locales, utilisation de la CPU / GPU... et de rappeler le support de plus en plus marqué de HTML5 sur les navigateurs du marché. La fin de l'année verra une déferlante, même si la spécification n'est pas prête. Mais HTML5 concerne aussi le téléphone et pas uniquement iPhone. Les démos de Gmail intégrant des morceaux de HTML5 ont été assez impressionnantes : drag and drop, notification.

## Des annonces attendues

On2, le codec vidéo racheté par Google, est aujourd'hui un des meilleurs formats, bien optimisé pour le web. Google le voit pour le strea-

ming. Il est aujourd'hui open source avec le projet WebM : VP8, sans licence à payer. WebM est l'addition de VP8 et de Vorbis. C'est une annonce attendue qui peut faire mal dans les 18 mois ! H.264 peut-il avoir du souci à se faire ? Les performances du codec impressionnent mais il faut attendre les outils et son adoption par les éditeurs. La bataille ne fait que commencer... Que vont faire les Apple et Microsoft ? Site web officiel : <http://www.webmproject.org/>

Simplifier l'usage des applications web avec HTML5 ou autre chose ? Il faut déjà trouver la bonne application, on perd du temps à chercher. Ensuite, le développeur doit créer les nouvelles applications, plus riches, plus simples. Google dévoile Chrome WebStore ! Google livrait déjà un webstore pour les Google Apps. Chrome a plus de 70 millions d'utilisateurs. Un marché potentiel important donc. Il sera compatible avec ChromeOS, pour les applications gratuites et payantes. Pas de date précise sur sa disponibilité. Une partie développeur sera, elle aussi, bientôt disponible.

## La portabilité des applications

Les applications web se multiplient à la maison, au bureau, sur le mobile. Et Google annonce dans la foulée un accord technique avec VMware. Le mot d'ordre : la portabilité des applications. Même si celle-ci reste très limitée dans un contexte Google - VMware. Le but ici est Java. Comme pour VMforce, VMware met en avant Spring et les outils CloudFoundry. De

quoi renforcer la position de VMware sur le cloud et améliorer l'exécution applicative pour Google AppEngine et de GWT sur le cloud. Ce dernier est prêt pour le mobile, le portable et le desktop ! Google aime tellement Apple, que les démos de GWT mobile se font sur un iPad...

Google App Engine for Business, autre annonce importante de Google I/O, on bénéficie d'une console domaine, d'un support pro, d'un SLA, de SSL et SQL et d'une nouvelle tarification (8 \$ par mois et utilisateur, plus l'application). App Engine passe enfin à la vitesse supérieure ! Mais attention tout ne sera pas disponible en même temps et tout cela s'échelonnera sur plus de 6-8 mois...

## Le nouvel Android est arrivé...

Google a profité de la conférence pour dévoiler Android 2.2. Cette version se focalise sur les performances. On bénéficie de nouvelles API pour le verrouillage du téléphone, la gestion du terminal. Côté Internet, Le moteur V8 (JavaScript) va permettre d'améliorer la vitesse d'exécution et d'affichage. L'ensemble des outils et SDK ont été mis à jour. C'est aussi l'arrivée de Flash Player 10.1 sur Android qui marque les esprits. L'autre grosse nouveauté est l'annonce de Google TV. Il s'agit d'une plateforme ouverte pour bénéficier du web sur la télévision et de l'accès aux contenus. L'environnement est basé sur Android et Chrome pour la navigation. Flash sera le module de lecture. Pour les développeurs, des API spécifiques seront disponibles.



# INSPIREZ! INFLUENCEZ! IMPRESSIONNEZ!



Vous possédez les données, mais vous n'avez ni le temps, ni le budget, ni le personnel pour présenter ces informations importantes d'une manière qui va impressionner. Avec NetAdvantage pour Silverlight Data Visualization d'Infragistics, vous pouvez créer sur le Web des visualisations de données et de tableau de bord avec Microsoft Silverlight (et bientôt WPF). Tout ceci va non seulement impressionner les décideurs, mais aussi faciliter leur prise de décision. Allez sur [infragistics.com/sldv](http://infragistics.com/sldv) aujourd'hui et trouvez l'inspiration pour créer des Killer Apps.

**Infragistics**  
KILLER APPS. No Excuses.

Infragistics Ventes France 0800 667 307  
Infragistics Europe Ventes +44 (0) 800 298 9055  
Infragistics India +91-80-6785-1111

Copyright 1996-2010 Infragistics, Inc. All rights reserved. Infragistics and the Infragistics logo and NetAdvantage are registered trademarks of Infragistics, Inc.

"Nous avons voulu nous lancer pour créer un véritable webzine musical, et c'est vers 1&1 que la plupart de nos amis dans le milieu nous ont conseillé de nous orienter. Nous avons pu créer un véritable site interactif grâce aux nombreuses offres que propose l'hébergeur. Aujourd'hui nous sommes suivis comme jamais nous l'avions espéré et c'est grâce à 1&1 que nous avons pu nous lancer dans cette aventure."

Amaury MARTIN B., [www.wearethemascotte.fr](http://www.wearethemascotte.fr)



# 10 ANS DE SU

\*Offre spéciale « 10 ans » : tous les Packs Hébergement 1&1 sont à 10 € HT/an la première année, soit 11,96 € TTC, frais de mise en service inclus ! A partir de la 2<sup>e</sup> année, leur prix varie de 2,38 € TTC/mois (Pack Perso Initial) à 35,87 € TTC/mois (WinPack Pro Performance). Offre soumise à un engagement de deux ans. Conditions détaillées sur notre site Internet. Offres sans engagement également disponibles.



# 10

## ANS À VOS CÔTÉS ...

... ça se fête ! Depuis le lancement de 1&1 France en 2000, nous nous sommes rapidement imposés comme un hébergeur de référence. Grâce à de solides bases et à une innovation constante, nous avons pu bâtir notre succès, qui est également devenu celui de nos clients. A l'occasion de nos 10 ans, nous avons décidé de VOUS faire profiter de ce succès. Rejoignez-nous maintenant et hébergez votre site pendant un an pour seulement 10 € HT !\*

**HÉBERGEZ VOTRE SITE WEB  
PENDANT UN AN  
POUR SEULEMENT**

# 10€\*

HT

Offre valable sur tous les packs hébergement !

**ÉCONOMISEZ  
JUSQU'À 418,48 €\***

## ACCÈS AVEC 1&1 !

The 1&1 logo, consisting of the text "1&1" in white on a blue square background.

[www.1and1.fr](http://www.1and1.fr)

# Maîtriser la diffusion de vidéo au format Flash

**P**our les développeurs, Flash rime de plus en plus avec la création de RIA et Flex, mais ce format n'en demeure pas moins un véritable standard de facto pour la vidéo en ligne. Si environ 80 % des contenus vidéo en ligne sont diffusés au travers de ce format, c'est en partie lié à la présence du lecteur Flash sur 98 % des postes connectés (PC, Linux ou Mac) mais aussi à sa flexibilité et aux possibilités de développement qui l'accompagnent. Nous allons revenir sur les 4 étapes clefs qui sont à considérer dans le processus de la mise en œuvre de la vidéo Flash : l'encodage, la création du lecteur vidéo, les méthodes de diffusion et la mise en production.

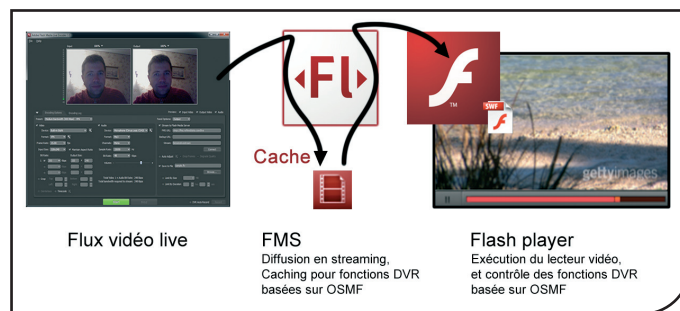
## Encodage

Contrairement à la plupart des lecteurs vidéo, le Flash Player ne dissocie pas les codecs du lecteur pour s'assurer que les utilisateurs n'ont rien à installer en plus pour lire les contenus. Il existe 3 codecs disponibles dans le lecteur Flash, à considérer selon ses projets pour obtenir des fichiers .F4V ou .FLV (video Flash) :

**Sorenson Spark** (depuis Flash 6) utilise un algorithme de décompression, certes ancien, et plus consommateur en bande passante que ses successeurs, mais moins gourmand en termes de ressources CPU. Au-delà de la restitution vidéo, le Lecteur Flash dispose uniquement de Sorenson Spark pour assurer de la compression vidéo temps réel sur les postes utilisateurs ; c'est la raison pour laquelle il est toujours utilisé pour le développement d'applications de communication temps réel. **On2 VP6** (depuis Flash 8) apporte de véritables gains en termes de bande passante et offre des possibilités graphiques avancées, notamment avec le support d'un canal alpha dans les vidéos (uniquement disponible avec on2 vp6, typiquement pour obtenir un effet de personnage détourné en premier plan) ; c'est le format à privilégier pour une intégration graphique avancée dans une page web ou dans un projet Flash.

**H.264** (depuis Flash 9.0.115) offre le meilleur ratio qualité/poids et permet de mettre à profit un des codecs les plus populaires du moment avec tous les bénéfices interactifs de Flash. Enregistrables en .FLV, vous pouvez également sauvegarder vos contenus H.264 en .F4V pour respecter des spécifications d'un format vidéo standardisé (ISO 14496-12) exploitable avec le lecteur Flash. La nouvelle version du Player Flash (10.1) offre des améliorations en termes de performance sur la lecture des vidéos H.264, notamment grâce à une accélération hardware de l'affichage. C'est le format à considérer dans la plupart des besoins de diffusion vidéo à la demande, notamment pour la HD.

Les outils de montage Adobe proposent de sauvegarder au format .FLV et .F4V mais également d'ajouter visuellement des cuepoints (pour faciliter la navigation ou synchroniser la vidéo avec d'autres contenus tels que des sous-titres). La fonction speech-to-text (disponible dans Premiere Pro) permet de créer ces cuepoints automatiquement sur les .FLV/.F4V avec une analyse de la voix : idéal si l'on souhaite programmer dans le .swf un moteur de recherche basé sur les paroles pour naviguer dans la vidéo. Au-delà des outils vidéo Adobe, la plupart des solutions de montage sauvegardent en .FLV. Pour le traitement par lot, l'Adobe Media Encoder permet en batch de convertir tous les contenus vidéo existants selon des profils que l'on prédéfinit (l'Adobe Media Live Encoder, disponible gratuitement, compresse pour sa part les flux live selon ces différents codecs).



## Création du lecteur vidéo

Autre spécificité du lecteur Flash, il exécute des fichiers .swf (capables d'accéder aux .flv/.f4v) mais ne permet pas de lire directement des contenus vidéos encodés (sauf si le contenu .FLV est embarqué dans le fichier .swf, ce qui est à proscrire du fait de la perte de toute capacité de streaming ou de téléchargement progressif). La seconde étape consiste donc à créer son propre lecteur vidéo en Flash, et ainsi de disposer d'un contrôle complet sur son intelligence et son ergonomie. Si vos besoins sont basiques en termes de création de lecteur (pause, plein écran, saut, volume...), de nombreux composants existent, ne nécessitant qu'un reskinning, aussi bien dans Flash Builder 4 (ex Flex Builder), dans Flash Professional CS5 que dans Dreamweaver CS5 lors de l'intégration des .FLV et .F4V.

Pour des projets avancés, Flash Professional CS5 ou Flash Builder 4 vous permettent d'utiliser un ensemble d'objets et de méthodes ActionScript 3 dédiées aux contrôles vidéos, selon l'approche de diffusion retenue parmi les 2 possibles : la réutilisation du composant FLVPlayback et l'accès à environ 80 méthodes associées (fl.video.FLVPlayback) ; ou la programmation avec les objets NetConnection et NetStream de la lecture dans un objet vidéo :

```
package {
    import flash.display.MovieClip;
    import flash.net.NetConnection;
    import flash.net.NetStream;
    import flash.media.Video;

    public class myvideoplayer extends MovieClip {

        public function myvideoplayer() {
            // création d'un objet NetConnection,
            // Prévoir la détection d'événement associé
            var ncConnection:NetConnection = new NetConnection();
            // Réglé à null le paramètre de connexion pour du téléchargement progressif,
            // Pointé vers une source rtmp pour du streaming;
            ncConnection.connect(null);

            // Création d'un objet NetStream sur l'objet NetConnection
            var nsStream:NetStream = new NetStream(ncConnection);
            // Réglage de la taille de la mémoire tampon en seconde
            nsStream.bufferTime = 5;
        }
    }
}
```



```
// Attache et affiche le flux vidéo NetStream dans un objet vidéo
var myvideo:Video = new Video();
myvideo.attachNetStream(nsStream);
addChild(myvideo);

// Lecture de la vidéo
nsStream.play("http://localhost/video.flv"); }
}
```

Parmi les spécificités de Flash, la capacité de contrôler dynamiquement un buffer permet d'adapter intelligemment les performances de lecture (en fonction de la détection de bande passante par exemple). Depuis Flash 9, il est également possible de basculer la vidéo en plein écran, en ajustant une propriété du stage : `Stage.displayState = "FULLSCREEN"` (par défaut réglé à `"NORMAL"`) ; cela nécessite également un ajustement dans la mise en production lors de l'intégration HTML.

L'objet `NetStream` permet d'identifier les cuepoints sur le flux vidéo qui peuvent être de deux types : 'navigation' pour accéder à des instants spécifiques de la vidéo (avec un menu interactif sur le modèle d'un DVD par exemple) ou 'événement' pour déclencher une interaction à un moment précis (popup d'informations ou commentaires en premier plan par exemple). Chaque cuepoint, au-delà d'être référencé par rapport à un timing sur la vidéo, dispose d'un paramètre pour ajuster l'interactivité (texte pour définir un sous-titre, la fonction à exécuter...). Les cuepoints étant encodés dans les fichiers vidéos, ils ne sont pas dynamiques : si vous souhaitez permettre l'ajout de marqueurs temporels dynamiquement par les utilisateurs de votre lecteur (échanges sociaux, ajout de commentaires dans la vidéo par exemple), préférez une gestion xml en parallèle des cuepoints.

Afin de faciliter le développement de lecteur vidéo avec la technologie Flash, Adobe a créé le framework opensource OSMF (Open Sources Media Framework, en version 0.95 actuellement, téléchargeable depuis <http://opensource.adobe.com/wiki/display/osmf/Downloads>), fournissant un ensemble de classes actionscript 3, exploitables aussi bien avec Flash Professional que Flash Builder. L'OSMF fournit également des bibliothèques de codes actionscript côté serveur (exploitable avec Flash Media Server) pour des options avancées dans des contextes de streaming (par exemple, fonctions de time-shifting ou DVR pour l'enregistrement côté serveur de flux vidéo live, enchaînement de média avec pre buffering, streaming dynamique selon la bande passante disponible). Cette couche d'abstraction simplifie considérablement la gestion et la diffusion des contenus, en considérant les flux images, audio et vidéo, streamées ou non, comme des objets médias ; elle dispose également d'une architecture extensible via des plugins (exploitation de contenus SMIL par exemple ou intégration directe avec akamai sans développement spécifique). L'exemple suivant illustre l'utilisation d'OSMF pour créer un lecteur vidéo avec une playlist et l'inclusion d'images.

```
package {
    import flash.display.MovieClip;

    //identifier dans les paramètres actionscript 3 de publication,
    //le répertoire OSMF avec le framework et les composants OSMF.swc
    import org.osmf.containers.MediaContainer;
```

```
import org.osmf.elements.VideoElement;
import org.osmf.media.MediaPlayer;
import org.osmf.media.URLResource;
import org.osmf.elements.SerialElement;
import org.osmf.elements.DurationElement;

public class mywebplayer extends MovieClip
{
    public function mywebplayer()
    {
        //définition des sources medias videos, images, sons ou swf
        var media1:VideoElement = new VideoElement( new URLResource( "v1.flv" ));
        var media2:VideoElement = new VideoElement( new URLResource( "v2.flv" ));
        var media3:ImageElement = new ImageElement( new URLResource( "il.jpg" ));

        //définition de la séquence de lecture qui peut être gérée dynamiquement,
        //Parallèlement permet de créer une séquence d'exécution concurrente
        //pour un affichage de publicité en premier plan plutôt qu'en insert
        var composition:SerialElement = new SerialElement();
        composition.addChild(media1);
        composition.addChild(new DurationElement(5, media3)); //
        image pendant 5s
        composition.addChild(media2);

        //Création d'un lecteur video sur la liste serialisée
        //l'Objet mediaplayer dispose des méthodes classiques play,
        seek, pause
        var player:MediaPlayer = new MediaPlayer();
        player.media = composition;

        //Affichage du lecteur
        var display:MediaContainer = new MediaContainer();
        display.addMediaElement( composition );
        addChild( display );
    }
}
```

Dans le cas d'une diffusion streamée, il est nécessaire d'utiliser la classe `org.osmf.net.netloader` pour identifier le flux rtmp du serveur FMS dans la construction de l'objet `VideoElement`.

```
myStream:String = "rtmp://localhost/vod/video.flv";
var player:MediaPlayer = new MediaPlayer();
player.source = new VideoElement(new NetLoader(), new URL
Resource(myStream));
```

À noter que le flux peut également être une source vidéo live : l'utilisation de la classe `org.osmf.net.dvr` permet d'avoir un mécanisme de caching côté serveur pour, à tout moment, permettre aux utilisateurs de revenir en arrière dans la diffusion du flux direct, et ce, sans pour autant avoir à programmer toute la logique applicative côté serveur. Gardez à l'esprit que la vidéo est un contenu parmi les autres dans Flash. Pour les utilisateurs connectés simultanément avec FMS, vous pouvez également envisager en actionscript 3 des interactions temps réel en premier plan des contenus: cela renforcera l'impact des médias consultés au travers d'expériences riches et immersives.

## Méthodes de diffusion

L'approche la plus simple à mettre en œuvre est l'emploi du téléchargement progressif : les vidéos sont hébergées sur un serveur web et le lecteur vidéo développé en Flash y accède en HTTP.

Cette méthode est populaire car elle limite l'usage de technologie serveur avancée, elle convient bien à la diffusion à la demande de vidéo de courte durée, non protégées (l'ensemble des contenus FLV transitant par le cache des navigateurs sur les postes utilisateurs) et sans besoin spécifique de qualité de service (QoS avec ajustement dynamique selon la détection de la bande passante dynamique).

La seconde approche est de recourir à une diffusion streamée des vidéos via le protocole rtmp (rtmps pour sa version sécurisée en ssl, rtmpt pour sa version tunnelée en http, ou rtmpe pour sa version encryptée).

Le lecteur vidéo initie une connexion persistante avec le serveur : cela assure une sécurisation des flux avec des échanges mémoire à mémoire (entre le serveur et le lecteur Flash) et offre la possibilité de sauter à un moment précis de la vidéo (sans attendre le téléchargement du contenu jusqu'à ce moment précis, grâce à un rechargement dynamique de la mémoire tampon).

Adobe Flash Media Server (FMS) est la solution de référence qui permet d'exploiter toutes les spécificités du protocole rtmp et qui, via une programmation ActionScript côté serveur, définit des comportements avancés de diffusion (l'OSMF permet d'en simplifier l'exploitation, le framework disposant également de bibliothèque de codes prédéfinis côté serveur pour FMS).

FMS permet en plus de streamer des vidéos à la demande, d'assurer une diffusion de flux vidéo live (l'encodage est le plus souvent géré par l'Adobe Media Live Encoder avec tous les codecs).

L'objet NetConnection permettant une communication bidirectionnelle entre le Flash Player et FMS, il est également possible de développer sa propre application Flash pour assurer la capture vidéo (dans ce cas, le lecteur Flash ne peut assurer la compression vidéo qu'avec le codec Sorenson Spark) : cette possibilité est notamment populaire quand Flash est utilisé pour créer des applications de chat vidéo temps réel entre les utilisateurs :

```
function init_cam():void
{
    // creation d'un objet camera pour capter le flux video local
    var cam:Camera = Camera.getCamera();
    cam.setMode(640, 480, 30);

    // creation d'un objet video avec le flux de la camera attaché
    var videocam:Video = new Video(cam.width, cam.height);
    videocam.attachCamera(cam);

    // affichage de la video
    var video:MovieClip = new MovieClip();
    video.addChild(videocam);
    addChild(video);
}
```

L'envoi des flux vers le serveur s'effectue avec NetConnexion et NetStream : ce dernier permet d'initier un flux unique montant (.play pour la lecture) ou descendant (.publish pour l'émission et .attachCamera pour lier le flux à l'objet NetStream) vers le serveur.

## Mise en production

Dans un contexte d'utilisation web avec une page HTML, l'intégration Flash s'effectue via les balises <object> et <embed>. Si vous avez implémenté dans votre lecteur des fonctions pour passer la vidéo en plein écran, vous devez ajouter le paramètre allowFullScreen. Si vous avez employé le codec on2 VP6 et ses propriétés de canal alpha, il est possible d'obtenir également une transparence de la vidéo en premier plan de la page (si le swf est positionné sur un calque HTML) : pour cela, il est nécessaire d'ajuster le paramètre wmode avec la valeur transparente :

```
<object
  codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=10,1,52,0" width="640" height="480"
  align="middle">
  <param name="allowScriptAccess" value="sameDomain" />
  <param name="movie" value="flash/myvideoplayer.swf" />
  <param name="quality" value="high" />
  <param name="wmode" value="transparent" />
  <param name="bgcolor" value="#ffffff" />
  <param name="allowFullScreen" value="true" />

  <embed src="flash/myvideoplayer.swf" quality="high" wmode=
    "transparent" allowFullScreen="true" bgcolor="#ffffff" width
    ="640" height="480" align="middle" allowscriptaccess="sameDomain"
    type="application/x-shockwave-flash" pluginspage="http://www.
    macromedia.com/go/getflashplayer" />
</embed>
</object>
```

Adobe recommande également d'utiliser SWFObject 2.2 (bibliothèque javascript opensource téléchargeable sur <http://code.google.com/p/swfobject/>) pour l'intégration de fichiers Flash dans les pages HTML, notamment pour les scripts assurant la détection des versions du lecteur Flash installées et la gestion des mises à jour nécessaires (Flash Professional l'utilise notamment lors de sa publication HTML).

Si l'on considère le plus souvent la technologie Flash pour le web, elle est également exploitable hors du navigateur grâce à AIR (Adobe Integrated Runtime). Vous pouvez ainsi exploiter le développement de vos lecteurs Flash pour un usage en mode offline ou avec une intégration au poste de travail. Dans le cas d'un développement de lecteur vidéo Flash pour AIR 2.0, la sécurisation des contenus qui seront disponibles en mode déconnecté peut se faire par DRM (Flash Access Packager pour l'application de droits sur les contenus vidéo, rtmpe pour la transmission cryptée et Flash Access License Server pour la gestion des droits sur les vidéos diffusées).

Si l'ensemble des applications Flash en *progressive download* ne nécessite qu'un serveur web, la mise en production des technologies de streaming implique la gestion d'un serveur FMS. Il est disponible gratuitement en version developer (10 connexions maximum), mais nécessite une licence pour sa mise en production si vous souhaitez gérer vous-même son installation en hébergement dédié. Autre possibilité, vous pouvez recourir à l'un des partenaires CDN d'Adobe (Akamai, Level3, Limelight notamment) pour un hébergement mutualisé.

■ Lionel Lemoine

Adobe France - Responsable Avant-Vente



# Smooth Streaming : un streaming vidéo efficace

Smooth Streaming est une extension pour IIS 7.0 permettant l'envoi d'un flux vidéo potentiellement haute-définition (1080p) par le protocole HTTP. Son objectif est de permettre d'avoir un flux s'adaptant au débit et au matériel du client, en modifiant la qualité de la vidéo à la volée (à l'instar du serveur vidéo de Flash). De cette manière, la vidéo est visionnée sans interruption. L'utilisation du protocole HTTP permet de profiter de nombreux avantages liés à sa normalisation. Notamment l'utilisation des caches HTTP disponibles sur de nombreux serveurs et l'utilisation de l'infrastructure REST.

Cette technologie a été annoncée le 28 octobre 2008 mais a été testée lors des jeux olympiques d'été 2008 de Beijing. En France, nous avons pu la découvrir en 2009 lors du Tour de France et de Roland Garros.

## De quoi parle-t-on ?

Le Smooth Streaming n'est pas un format de compression. Techniquement, la vidéo initiale est fragmentée en séquences de 2 secondes. Ces séquences sont disponibles en différentes qualités. De cette manière, en fonction du débit ou des capacités de décodage du client, le serveur choisit les fragments idéaux à envoyer de manière à éviter que la vidéo s'arrête pour gérer un temps de traitement. De plus, et grâce au protocole HTTP, ces fragments peuvent se mettre en cache du côté serveur pour pouvoir être rediffusés rapidement à d'autres clients. Le Smooth Streaming, de par sa structure fragmentée, apporte des fonctionnalités intéressantes lors de directs. Avec notamment la possibilité de mettre en pause le flux, de faire des retours arrière et bien sûr de reprendre le direct de manière fluide. Pour préparer une vidéo au Smooth Streaming, nous pouvons utiliser pour le cas par cas l'outil Expression Encoder qui, dès le SP1 de sa version 2, gère l'encodage en Smooth Streaming ou bien automatiser le traitement par le code avec Expression Encoder 3 SDK.

## Encodage avec Expression Encoder.

Commençons par voir comment réaliser cette opération avec Expression Encoder 2 SP1. Attention, la version gratuite ne gère pas les fonctionnalités liées au Smooth Streaming. Après avoir lancé l'application, réalisons un glisser-déposer de notre

vidéo dans la fenêtre principale. Réglons les options du menu « Encoder ». Notre profil vidéo doit gérer une « diffusion adaptative en continu », le mode de sortie sera une « Diffusion en continu lisse » autrement dit du « Smooth Streaming ». La taille du Profil vidéo sera « Profil Adaptative » [Fig.1].

De la même manière, réglons nos paramètres de sorties. Si vous souhaitez avoir un aperçu, laissez la case « afficher un aperçu [...] » cochée. De cette manière, un navigateur se lancera à la fin de l'encodage avec le modèle de lecteur choisi. Nous pouvons aussi choisir de publier directement sur notre serveur IIS grâce à l'option « Publier sur » et en utilisant WebDav (WebDav devra être installé sur votre serveur). Pour finir nous lançons l'encodage en cliquant sur Fichier/Coder ou en utilisant le raccourci clavier Ctrl+E. Une fois le fichier encodé, vous retrouverez différents fichiers en .ismv dans le dossier de sortie que vous avez défini. Ces fichiers .ismv correspondent aux différentes qualités d'encodage de votre vidéo. Deux fichiers xml en .ism et .ismc se trouvent aussi dans ce dossier, ils décrivent la fragmentation et les formats utilisés. En fonction de votre choix au niveau de l'aperçu, vous pourrez retrouver un .xap et une page default.html qui l'héberge. C'est votre lecteur [Fig.2].

## Encodage avec Expression Encoder 3 SDK.

Voyons maintenant comment réaliser cette opération avec du code C#. Nous allons réaliser une application console basique pour montrer le fonctionnement. Attention, il existe des problèmes de compatibilité avec le Framework 4.0, nous travaillerons donc sur le Framework 3.5. Commençons par ajouter les références Microsoft.Expression.Encoder, Microsoft.Expression.Encoder.Types,

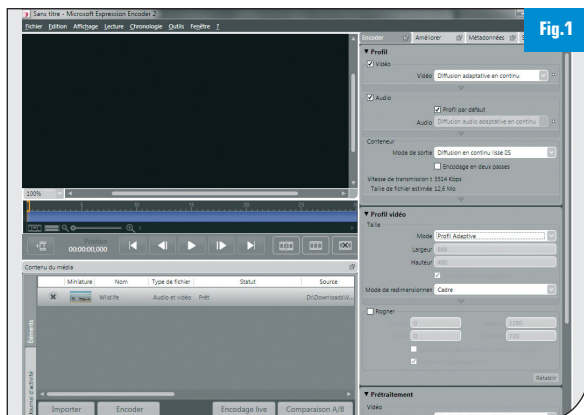


Fig.1



Fig.2

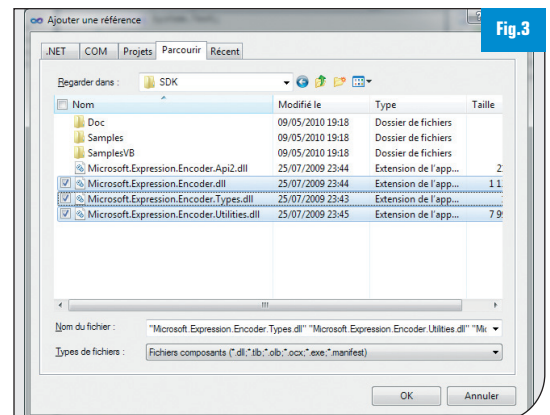


Fig.3

Microsoft.Expression.Encoder.Utilities disponibles par défaut dans le dossier **C:\Program Files\Microsoft Expression\Encoder 3\SDK** [Fig.3]. Nous devons aussi ajouter des références à System.Drawing et WindowsBase. Ajoutez ensuite le code suivant dans la fonction Main() de votre projet :

```
C#

//Localisation sur le disque dur
string path = @"D:\Downloads\Vidéos\Wildlife.wmv";
Job encoder = new Job();

//Définition de la source
AudioVideoFile source = new AudioVideoFile(path);

// Création du profil vidéo
AdvancedVC1VideoProfile videoProfile = new AdvancedVC1VideoProfile();

// Création des différents streams
//VariableConstrainedBitrate prend en paramètre la moyenne du
débit en bit/s et le débit maximal
//Le deuxième paramètre passé à la méthode Add correspond à la taille
de notre vidéo. Ici nous voulons garder la taille de la source
videoProfile.Streams.RemoveAt(0);
videoProfile.Streams.Add(new VariableConstrainedBitrate(1000,
1600), source.VideoStreams[0].VideoSize);
videoProfile.Streams.Add(new VariableConstrainedBitrate(600,
1600), source.VideoStreams[0].VideoSize);

// Nos réglages liés au Smooth Streaming
videoProfile.SmoothStreaming = true;
videoProfile.Streams.AutoSize = true;

// On règle les options autour du son
WmaAudioProfile audioProfile = new WmaAudioProfile()
{
    Bitrate = new VariableConstrainedBitrate(64, 96,TimeSpan.
FromSeconds(1.5)),
    Channels = 2,
    Codec = AudioCodec.Wma,
    BitsPerSample = 16
};

//On crée notre élément. A savoir que l'on peut ajouter
d'autres sources vidéos (media.Sources.Add(new Source...))
MediaItem media = new MediaItem(path)
```

```
{
    OutputFormat = new WindowsMediaOutputFormat() { VideoProfile =
        videoProfile, AudioProfile = audioProfile },
    ThumbnailMode = ThumbnailMode.BestFrame
};
// On ajoute nos items à notre encoder
encoder.MediaItems.Add(media);
// On spécifie le dossier de sortie et on lance l'encodage
encoder.OutputDirectory = @"C:\SmoothStreaming\";
encoder.Encode();
```

**Note :** Vous pouvez vous abonner aux événements EncodeProgress et EncodeCompleted pour être informé de la progression et de la fin de votre encodage.

## Côté serveur

Sur une plate-forme IIS 7.0 nous installerons le plugin Integrated Media Platform que nous trouverons sur le site de IIS : <http://www.iis.net/overview/IntegratedMediaPlatform>. Cela permettra d'apporter à notre serveur les fonctionnalités liées au Smooth Streaming. À savoir, la plate-forme pour la communication à travers HTTP, la possibilité de réaliser un flux en direct et l'outil Bit Rate Throttling qui mesure la vitesse de diffusion de notre média au lecteur d'un client. Assurez-vous de votre installation en sélectionnant un de vos sites Web dans le Gestionnaire de Serveurs, les fonctionnalités autour du Smooth Streaming devraient avoir fait leur apparition [Fig.4]. Notre serveur est prêt à diffuser du contenu en Smooth Streaming. Il nous suffit de déployer un lecteur et nos fichiers vidéo dans le gestionnaire de site web. Nous pouvons tester en copiant le dossier généré lors de l'encodage avec Expression Encoder (un lecteur est déjà intégré si vous avez sélectionné l'aperçu).

Nous pouvons aussi installer dans un répertoire web les exemples fournis par Microsoft : <http://go.microsoft.com/?linkid=9652408>. Le fichier *Big\_Buck\_Bunny\_Sample-IIS\_Smooth\_Streaming\_Beta.exe* permet de tester le Smooth Streaming sur la vidéo éponyme réalisée par la fondation Blender. Il met à disposition deux lecteurs, un troisième peut-être rajouté en installant dans le même dossier *Sample\_Client\_Refresh-IIS\_Smooth\_Streaming.exe*. Nous pouvons tester nos vidéos converties précédemment en déplaçant les fichiers .ism, .ismc et .ismv correspondant dans ce dossier. Il nous suffira d'appeler la vidéo souhaitée en entrant son nom.

Vous pouvez réaliser vos propres lecteurs en utilisant le Smooth Streaming Player Development Kit disponible à l'adresse suivante : <http://www.iis.net/download/smoothplayer>. Le projet Open Video Player (OVP) propose un lecteur Open Source capable de gérer le Smooth Streaming : <http://openvideoplayer.sourceforge.net/>.

## Conclusion

Grâce au Smooth Streaming, ce n'est plus le client qui doit s'adapter à la vidéo mais le flux qui s'adapte au client. Un confort important est donc gagné lors du visionnage. Nous avons donc vu comment encoder manuellement notre vidéo avec Expression Encoder mais aussi comment automatiser la procédure. En implémentant une gestion multi-threadée, grâce à ce SDK, nous pouvons facilement évoquer un scénario de traitement automatisé sur un serveur dédié. Il ne reste plus qu'à déployer un lecteur qui ira piocher dans les vidéos enregistrées.

■ Jean-Christophe Vasselon - Dotnet-France

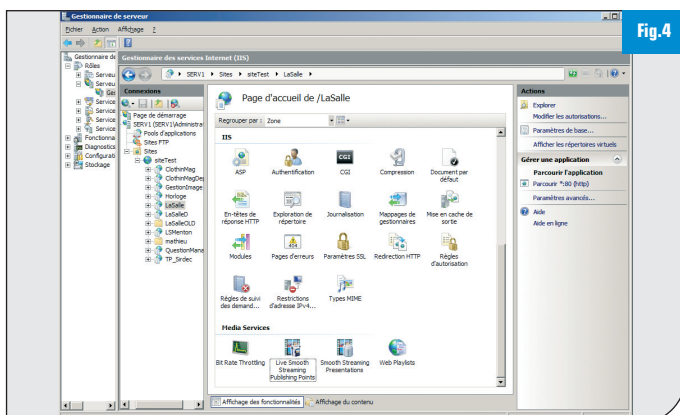


Fig.4



**ActiveReports 6** à partir de € 521

GrapeCity

Dernière version du générateur de rapports .NET sans droits le plus vendu.

- Moteur de génération de rapports rapide, souple avec licence sans droits pour le Web/Windows
- Visualisation des données et contrôles de mise en forme: Chart, Barcode et Table Cross Section
- Prise en charge de Medium Trust dans l'environnement ASP.Net
- Vaste gamme de formats d'exportation/aperçus: Windows Forms Viewer, Web Viewer, Flash & PDF
- Contrôle utilisateur de conception avec API souple et personnalisation

**FusionCharts** à partir de € 148InfoSoft Global  
Empowering human thought

Diagrammes interactifs et animés pour les applications Web et les applications de bureau.

- Animez vos applications Web avec les diagrammes Flash animés
- Créez des diagrammes compatibles AJAX pouvant changer côté client sans requêtes serveur
- Exportez les diagrammes en tant qu'images/PDF et les données en CSV pour les rapports
- Créez des jauges, des diagrammes financiers, de Gantt, en entonnoir et plus de 550 mappages
- Utilisé par plus de 15 000 clients et quelques 250 000 utilisateurs dans 110 pays

**Aspose.Words for .NET** à partir de € 670ASPOSE  
The .NET & Java Component Publisher™

Lisez, modifiez et écrivez des documents Word sans Microsoft Word.

- Création de documents, manipulation du contenu/formatage, puissante capacité de fusion de courrier et exportation en DOC/HTML
- Accès détaillé à tous les éléments d'un document par programmation
- Nouveau : support des régions de fusion d'e-mails incorporés et plus
- Support DOC, OOXML, RTF, HTML, OpenDocument, PDF, XPS et EPUB

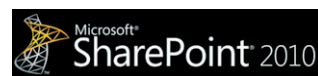
**InstallAware Studio** à partir de € 1 538

InstallAware

Une des meilleures solutions d'installation pour Windows Installer.

- Créer des programmes d'installation pour les FSI et les entreprises déployant des produits, des correctifs et des mises à jour sur le Web
- Données/logique d'installation personnalisées, variables de compilation
- Débogueurs visuels pour créer des fenêtres/messages personnalisés
- Aucune licence supplémentaire requise ; nombre illimité d'ordinateurs

# SharePoint 2010 : une version au top pour le développeur



Sharepoint 2010 (ou SP 2010) est disponible ! Cette version majeure de la plate-forme portail, collaborative, réseau social de Microsoft apporte de nombreuses nouveautés pour l'entreprise, l'administrateur mais aussi pour le développeur et particulièrement le développeur .Net et web. Explorons ensemble Sharepoint 2010 !

**A**près l'installation de SharePoint 2010, on peut découvrir le nouveau design qui rappelle beaucoup celui de la suite Office (dont Sharepoint reprend de nombreux principes d'interface) : [Fig.1]. L'interface est riche et se rapproche du RIA puisque SP2010 utilise un peu partout l'Ajax/Javascript et ... le Silverlight 3 ! Quel bonheur de voir que dans le menu "Action du site", la partie More Options est entièrement en Silverlight : [Fig.2].

Des nouvelles technologies clientes, une nouvelle master page, un ribbon qui a fait son apparition... SharePoint 2010 fait peau neuve et ce n'est pas juste pour faire joli. Il y a une bonne gestion des navigateurs puisque tout au long de mes tests j'ai utilisé Chrome et je n'ai pas eu de soucis de compatibilité.

SharePoint "14" (Office 2010 et consorts font partie du projet nom de code Wave 14) a été renommé en deux versions :

- Microsoft SharePoint Server 2010, anciennement MOSS 2007
  - Microsoft SharePoint Foundation 2010, anciennement WSS v3
- Comme en 2007, SharePoint Server est la version payante qui s'appuie sur la version gratuite SharePoint Foundation, elle-même basée sur ASP.NET 3.5 SP1 et .NET en général.

## SharePoint Designer 2010

SharePoint Designer est un outil type Expression Web qui permet de modifier les pages d'un site SharePoint, créer des petits workflows ou encore administrer quelques parties d'un site. L'un des principaux défauts de cet outil est dû au fait qu'il ne s'intègre pas dans un cycle de vie de développement et cela devient vite compliqué de réa-

liser un système de versionning / gestion de code source quand nous l'utilisons. Cependant, il reste bien utile et fait maintenant partie du menu "Site Actions", ce qui rend son utilisation un peu plus accessible à l'utilisateur lambda. L'apparence a un petit peu changé, SharePoint Designer sera, entre autres, utile lors de développements de WebPart type XSLT List View [Fig.3].

## Premier pas avec Visual Studio 2010

Le modèle de développement SP 2010 est basé sur .Net et Visual Studio 2010 [Fig.4].

Ces outils apportent quelques nouveautés comme le SharePoint Explorer qui s'ajoute au Server Explorer. Ce plug in centralise les informations de votre SharePoint dans Visual Studio [Fig.5]. Un projet sous Visual Studio 2010 se représente de cette manière : [Fig.6].

Une nouveauté est le dossier "features" qui représente les features qui vont être déployées. Ce dossier s'accompagne d'un assistant pouvant configurer toutes les features (Les items, les dépendances, le scope...) [Fig.7]. Et côté pratique, VS2010 attache automatiquement le débogueur au processus W3WP : [Fig.8].

## Définition de liste et Event Handler

Les listes SharePoint 2010 ont fait des progrès ! Tout d'abord SP2010 est capable d'accueillir des listes de très grande taille. Les listes peuvent, entre autres, avoir une colonne unique, faire des lookup interlistes, gérer l'intégrité des données... En résumé, on a enfin un vrai modèle relationnel.

## Les Event Handler en SharePoint 2007....

Les event handler sont des événements qui se déclenchent de façon synchrone ou asynchrone. Ce sont un peu comme des Workflows sans interface graphique (ou juste une page d'erreur non personnalisée) et qui

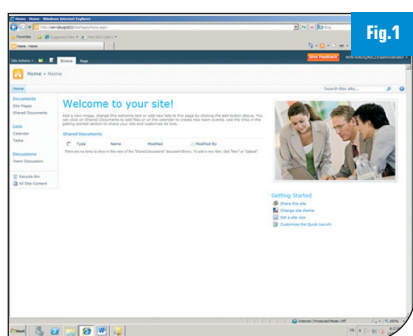


Fig.1

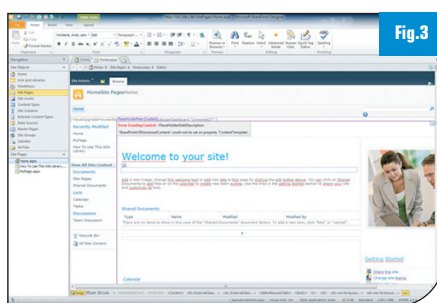


Fig.3

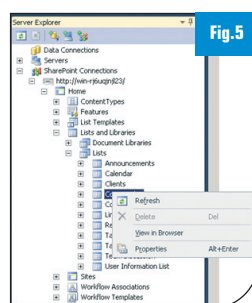


Fig.5

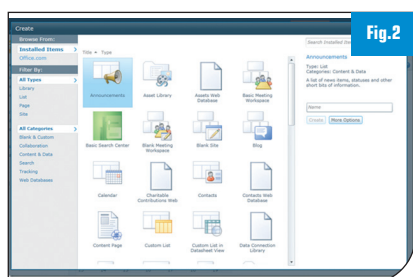


Fig.2

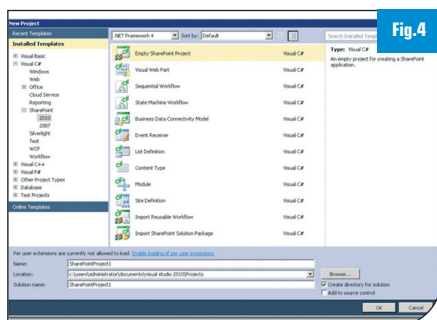


Fig.4

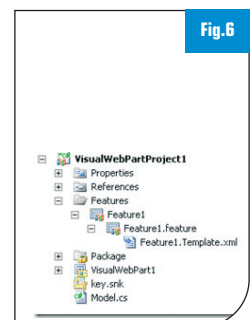


Fig.6

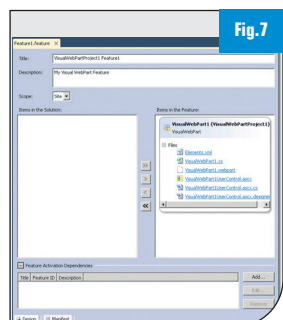


Fig.7



effectuent des actions plus simples. Les événements synchrones démarrent dès que l'action est réalisée et ne rendent la main à l'utilisateur que quand votre code est terminé. Les événements asynchrones se lancent après que l'action soit réalisée, il est non bloquant mais ne permettait pas d'annuler l'action faite par l'utilisateur.

#### En SharePoint 2010...

Les event handler permettent grâce à une classe Synchronization de pouvoir faire un "rollback" avec un événement asynchrone. Ils sont aussi capables d'afficher une page personnalisée. De nouveaux événements ont fait leur apparition essentiellement sur les listes de données :

- ListAdding / List Added
- ListDeleting / ListDeleted

Exemple : capter la destruction d'une liste avec l'événement ListDeleting et afficher une page personnalisée [Fig.9].

Un assistant s'ouvre pour vous demander sur quel type d'événement vous souhaitez vous abonner : [Fig.10].

On s'abonne à ListDeleting (donc en synchrone) et on va arrêter l'action de suppression :

```
view plaincopy to clipboardprint
1. public override void ListDeleting(SPListEventProperties properties)
2. {
3.     base.ListDeleting(properties);
4.     properties.Cancel = true;
5.     properties.Status = SPEventReceiverStatus.CancelWithRedirectUrl;
6.     properties.RedirectUrl = "/_layouts/EventReceiverProject2/Error.aspx";
7. }
```

On crée la page personnalisée : [Fig.11].

## Développeur Dashboard

Le développeur Dashboard est un outil pour détecter ce qui consomme de la ressource sur votre SharePoint. On l'active de cette manière (ne pas oublier le IISRESET ensuite) : [Fig.12]. Et on voit apparaître sur le SharePoint une petite icône : [Fig.13].

Lorsqu'on active le dashboard, un panel fait son apparition en bas de page avec un détail sur le temps passé entre chaque appel de fonction, de requêtes ou d'allocations mémoire.

## Création d'une webpart

Les Web parts fonctionnent globalement comme en SharePoint 2007. L'apparition du template Visual WebPart n'est ni plus ni

moins ce que l'on faisait déjà en SharePoint 2007. Je ne décrirai pas ici tout ce qui est persistance de données d'une WebPart ou encore les WebParts connectées puisque le code est le même.

Allons plus loin avec un petit exemple , après avoir créé un nouveau projet de type "Visual WebPart" un assistant s'ouvre : [Fig.14].

Il faut mettre l'adresse du SharePoint concerné et laisser le choix par défaut, nous y reviendrons plus tard.

Le projet généré possède différents fichiers. Le fichier Elements.xml qui représente les fichiers embarqués dans la solution et le fichier .webpart qui contient des informations sur la Web Part comme son titre ou sa description. VisualWebPart1.cs contient une classe simple qui fait appel à Page.LoadControl pour charger le contrôle utilisateur.

```
view plaincopy to clipboardprint
1. [ToolboxItemAttribute(false)]
2. public class VisualWebPart1 : WebPart
3. {
4.     private const string _ascxPath = @"~/_CONTROLTEMPLATES/VisualWebPartProject1/VisualWebPart1/VisualWebPart1UserControl.ascx";
5.
6.     public VisualWebPart1()
7.     {
8.     }
9.
10.    protected override void CreateChildControls()
11.    {
12.        Control control = this.Page.LoadControl(_ascxPath);
13.        Controls.Add(control);
14.        base.CreateChildControls();
15.    }
16.
17.    protected override void RenderContents(HtmlTextWriter writer)
18.    {
19.        base.RenderContents(writer);
20.    }
21. }
```

Le fichier ascx représente le code ASPNET et le .cs son code behind. Lors de la compilation, un fichier WSP est généré, la feature est déployée et le navigateur s'ouvre sur la création d'une page pouvant accueillir une Web Part. Il existe aussi les XSLT List View Web

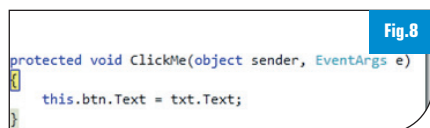


Fig.8

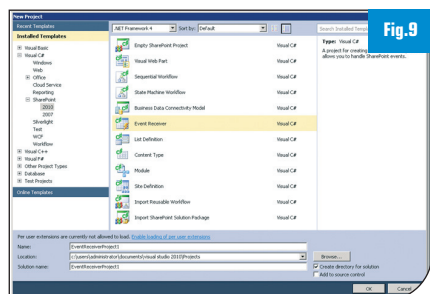


Fig.9

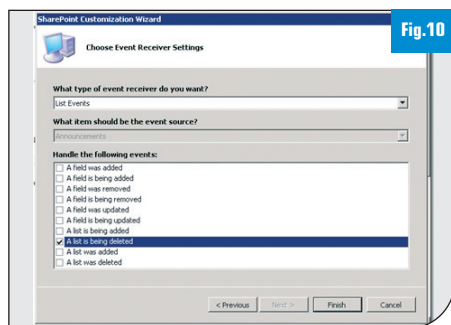


Fig.10

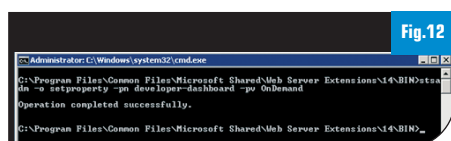


Fig.12

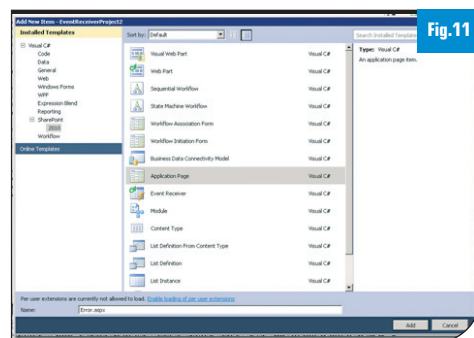


Fig.11

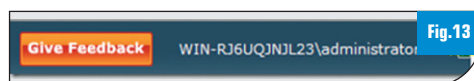


Fig.13

Part qui remplacent la Data View Web Part de WSS 3.0. Ce type de webpart, qui affiche des données d'une liste, se fait très facilement avec SharePoint Designer : [Fig.15].

Cela fait appel à un contrôle ASP.NET appelé XsltListViewWebPart et qui attend une source de données, les champs à afficher et éventuellement une requête de filtrage.

## Manipuler le ribbon

Le ribbon (ou ruban) nous vient directement d'Office et est l'une des nouveautés majeures. Il facilite grandement la navigation dans les projets et l'environnement Sharepoint [Fig.16].

Il est possible de le personnaliser par le code, ce qui est bienvenu ! Exemple : Créer un groupe de contrôle dans le ribbon contenant deux boutons.

On crée un projet vide. On ajoute un élément vide : [Fig.17].

Et on ajoute ainsi des balises au fichier Elements.xml. Le but est de créer un groupe de boutons dans l'onglet "List" d'une custom list.

```
view plaincopy to clipboardprint
```

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <Elements xmlns="http://schemas.microsoft.com/sharepoint/">
3.   <CustomAction Id="Ribbon.List.GroupAction"
4.     Location="CommandUI.Ribbon"
5.     RegistrationId="100"
6.     RegistrationType="List">
7.     <CommandUIExtension>
8.       <CommandUIDefinitions>
9.         <CommandUIDefinition Location="Ribbon.List.Groups.
10.           _children">
11.             <Group Id="Ribbon.List.MyGroup"
12.               Title="MyGroup"
13.               Template="Ribbon.Templates.Flexible2"
14.               Sequence="10001">
15.               <Controls Id="Ribbon.List.MyGroup.Controls">
16.                 <Button Id="Ribbon.List.MyGroup.Button1"
17.                   Command="Button1Command"
18.                   LabelText="Button1"
19.                   TemplateAlias="o1" />
20.                 <Button Id="Ribbon.List.MyGroup.Button2"
21.                   Command="Button2Command"
22.                   LabelText="Button2"
23.                   TemplateAlias="o2" />
24.               </Controls>
25.             </Group>
26.           </CommandUIDefinition>
27.         </CommandUIDefinitions>
28.       </CommandUIExtension>
29.     </CustomAction>
30.   </Elements>
```

```
27.       </CommandUIDefinition>
28.       <CommandUIDefinition Location="Ribbon.List.Scaling.
29.         _children">
30.         <MaxSize Id="Ribbon.List.Scaling.MyGroup.MaxSize"
31.           GroupId="Ribbon.List.MyGroup"
32.           Size="LargeMedium" />
33.       </CommandUIDefinition>
34.     </CommandUIDefinitions>
35.   <CommandUIHandlers>
36.     <CommandUIHandler Command="Button1Command"
37.       CommandAction="" />
38.     <CommandUIHandler Command="Button2Command"
39.       CommandAction="" />
40.   </CommandUIHandlers>
41. </CommandUIExtension>
42. </CustomAction>
43. </Elements>
```

Reprenons en détail ce code. Dans un premier temps nous déclarons un CustomAction (comme en SharePoint 2007) qui va nous permettre d'aller modifier un des menus. Nous précisons à ce CustomAction l'ID Ribbon.List.XXX. Cela signale que l'on travaille sur le menu de type ribbon et son sous-menu "List". RegistrationId correspond au type de liste sur laquelle on veut voir apparaître le menu (100 = CustomList). RegistrationType applique le menu pour les listes (on aurait pu choisir ContentType, FileType ou ProgId). La balise CommandUIDefinition contient un groupe de contrôle (deux boutons). Notez la présence de TemplateAlias sur les boutons qui correspond au type d'affichage. Pour finir les CommandUIHandler permettent de capter les événements du code JavaScript ou C#. C'est ici qu'on aurait pu lier notre code à une redirection.

## Utiliser le dialog framework

Le dialog Framework vous donne la possibilité de créer des fenêtres JavaScript modales de ce type : [Fig.18].

Pour cela plusieurs options, entre autres, celle de déclencher du JavaScript dans le CommandAction d'un bouton du ribbon. On reprend donc l'exemple fait précédemment et on implémente le clic sur le bouton. Pour cela on ajoute déjà une page à notre projet (Add->New item sur le projet) : [Fig.19]. Et c'est cette page que l'on va vouloir afficher dans une modal popup.

```
view plaincopy to clipboardprint
```

```
1. <CommandUIHandler Command="Button2Command"
2.   CommandAction="javascript:
3.     
```

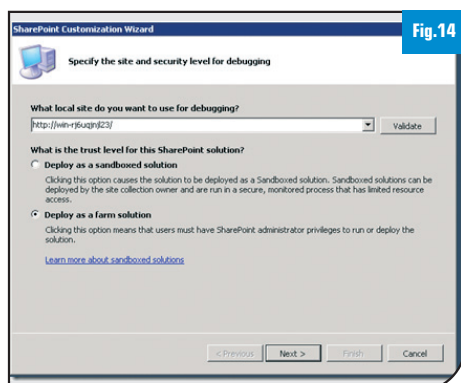


Fig.14

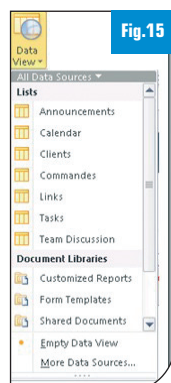


Fig.15

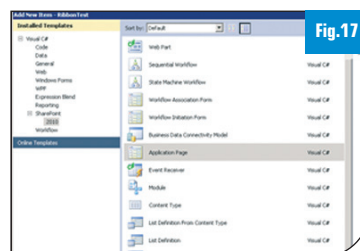


Fig.17

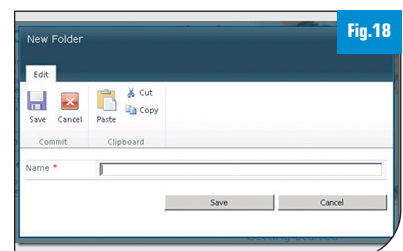


Fig.18

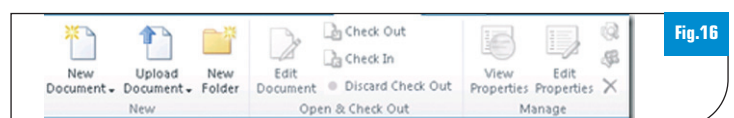


Fig.16



```

4. function Callback(dialogResult, returnValue)
5.     {
6.         SP.UI.Notify.addNotification('Popup fermée');
7.     }
8.
9. var options = {
10.    url: '/_layouts/RibbonTest/MaPage.aspx',
11.    title: 'Custom Dialog Box',
12.    allowMaximize: false,
13.    showClose: true,
14.    width: 400,
15.    height: 400,
16.    dialogReturnValueCallback: Callback };
17.
18. SP.UI.ModalDialog.showModalDialog(options); " />

```

On aurait aussi pu utiliser la méthode RefreshPage pour rafraîchir les données de la liste. Le callback sert à afficher une petite notification à l'utilisateur à la fermeture de la popup : [Fig.20].

## La Status Bar

Le status bar est utilisable comme les notifications grâce au JavaScript.

```

view plaincopy to clipboardprint
1. SP.UI.Status.addStatus('Popup fermée');
[Fig.21]

```

À noter que l'on peut changer la couleur avec la méthode setStatusPriColor ou encore modifier le status avec updateStatus et le supprimer avec removeStatus.

## Linq To SharePoint

Linq To SharePoint ou SPLinq est un outil de requêtage qui offre une alternative à la génération à la main de requête CAML avec SPQuery (limité au contexte du serveur). On connaît déjà les avantages de Linq, facile d'utilisation, vérification à la compilation des erreurs, IntelliSense, une seule et même façon de requêter des données (SQL, XML...)... SPLinq ne remplace pas SPSiteDataQuery qui permet de réaliser des requêtes sur toute une collection de sites. SPLinq utilise donc la syntaxe de Linq pour générer des requêtes CAML. Comme Linq-To-SQL qui utilise SQLMetal, là on utilisera SPMetal pour générer un modèle par rapport à un site SharePoint [Fig.22]. Cette commande nous génère un fichier Model.cs qu'il faut ensuite incorporer à notre projet. Ce fichier contient le contexte de données de votre SharePoint comme par exemple les listes person-

nalisées. Pour utiliser SPLinq, le projet doit aussi référencer la dll Microsoft.SharePoint.Linq et utiliser ces espaces de noms :

```

view plaincopy to clipboardprint
1. using Microsoft.SharePoint.Linq;
2. using System.Linq;

```

[Fig.23]

Pour illustrer mon exemple, j'avais créé une première liste "entreprise" et une liste "commandes":

```

view plaincopy to clipboardprint
1. protected void ClickMe(object sender, EventArgs e)
2.     {
3.         ModelDataContext context = new ModelDataContext(SP
Context.Current.Web.Url);
4.         EntityList<CommandesItem> commandes =
5.         context.GetList<CommandesItem>("Commandes");
6.
7.         var query = from commande in commandes
8.                     where commande.Entreprise.Title.Equals
("DOLLON")
9.                     select commande;
10.
11.         spGridView.DataSource = query;
12.         spGridView.DataBind();
13.
14.     }

```

Pour connaître le CAML généré, vous pouvez utiliser cette méthode. Comme les autres providers Linq, SPLinq propose de manipuler les données de façon CRUD (Create, Read, Update, Delete). Nous avons vu "read" ci-dessus, passons au reste :

### • Create

Comme pour l'Entity Framework, la création se passe avec l'instanciation d'une nouvelle entité, l'appel à la méthode Insert et l'envoi du "commit" avec la méthode SubmitChanges();

```

view plaincopy to clipboardprint
1. ModelDataContext context = new ModelDataContext(SPContext.
Current.Web.Url);
2. EntityList<ClientsItem> clients = context.GetList<Clients
Item>("Clients");
3.
4. ClientsItem client = new ClientsItem();
5. client.Entreprise = "Exakis2";

```

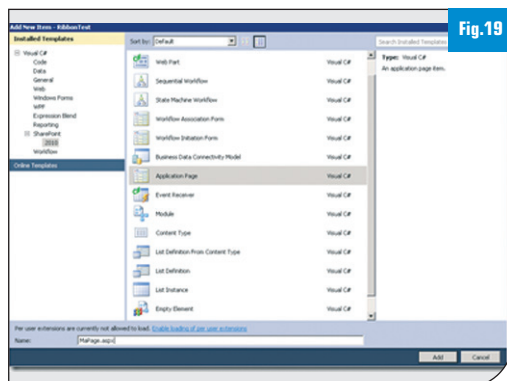


Fig.19

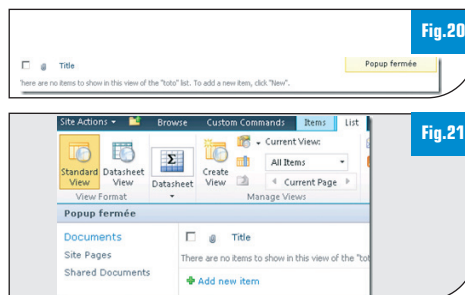


Fig.20

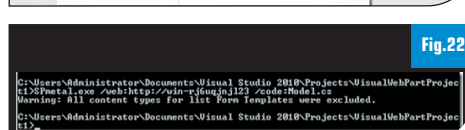


Fig.21



Fig.22

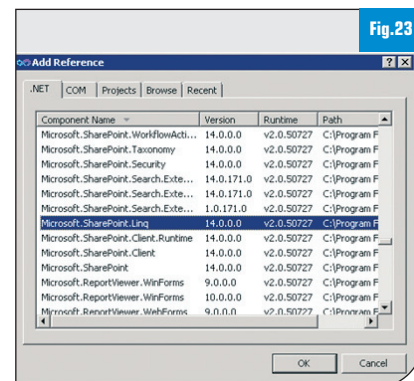


Fig.23

```
6. client.Title = "DOLLON2";
7. clients.InsertOnSubmit(client);
8. context.SubmitChanges();
```

Pensez à gérer les exceptions de type conflit ou duplication.

## • Update

La mise à jour se fait en retrouvant l'entité concernée et en modifiant ses attributs.

```
view plaincopy to clipboardprint
1. ModelDataContext context = new ModelDataContext(SPContext.
Current.Web.Url);
2. EntityList<ClientsItem> clients = context.GetList<Clients
Item>("Clients");
3.
4. var client = (from cl in clients
5.               where cl.Title.Equals("DOLLON2")
6.               select cl).First();
7.
8. client.Title = "VALENTIN";
9. context.SubmitChanges();
```

## • Delete

Pour la suppression, il y a deux moyens. Le premier étant de retrouver l'entité concernée comme pour la modification et effectuer la suppression.

Cette méthode n'est pas conseillée car elle vous impose de charger toute l'entité avant de la supprimer.

Ce qui peut se révéler particulièrement lourd si vous avez à supprimer des milliers de lignes. Le second moyen est de retrouver seulement l'ID de la personne à supprimer. Exemple :

```
view plaincopy to clipboardprint
1. ModelDataContext context = new ModelDataContext(SPContext.
Current.Web.Url);
2. EntityList<ClientsItem> clients = context.GetList<Clients
Item>("Clients");
3.
4. var id = (from cl in clients
5.           where cl.Title.Equals("VALENTIN")
6.           select cl.Id).First();
7.
```

```
8.
9. ClientsItem client = new ClientsItem();
10. client.Id = id;
11.
12. context.Clients.Attach(client);
13. context.Clients.DeleteOnSubmit(client);
14.
15. context.SubmitChanges();
```

## Le Client Object Model pour CLR

Le Client OM est un outil pour développeur permettant d'accéder au modèle objet de SharePoint (SPServer, SPFarm...). Ce modèle contenu dans la dll Microsoft.SharePoint.Client.dll est décliné en plusieurs versions comme pour la CLR, Silverlight mais aussi JavaScript. Dans ce chapitre nous aborderons la version .NET en créant une petite application WPF qui travaillera sur un SharePoint distant. Vous noterez la simplicité d'utilisation comparée à l'utilisation des Web Services de SharePoint 2010 où il fallait formater un fichier XML à envoyer au serveur pour modifier un élément d'une liste. Voici un récapitulatif des "armes" à disposition du développeur pour utiliser les objets SP : [Fig.24].

Voilà comment fonctionne le Client Object Model (toujours en rouge le côté client et en vert le côté serveur) : [Fig.25].

Important à préciser : le Client OM fonctionne à partir du framework 3.5. Il est donc utilisable dans de nombreux projets (Winforms, WPF, Office, Application console, service WCF...). La DLL se trouve dans le dossier ISAP :

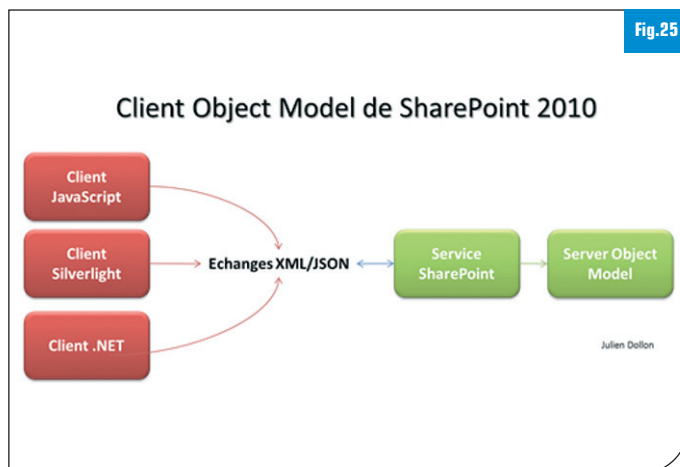
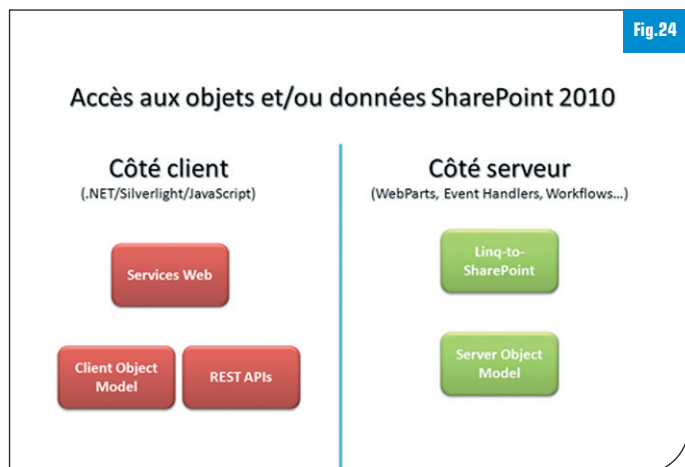
```
view plaincopy to clipboardprint
1. C:\Program Files\Common Files\Microsoft Shared\Web Server
Extensions\14\ISAPI
```

On crée un projet WPF et on y ajoute ces deux références : [Fig.26].

```
view plaincopy to clipboardprint
1. using Microsoft.SharePoint.Client;
2. using SP = Microsoft.SharePoint.Client;
```

L'interface graphique de notre projet ressemble à ceci : [Fig.27].

Le but étant d'implémenter les boutons en traitant la liste Clients vue dans un chapitre précédent. Le premier bouton qui charge une liste possède ce code :





```
view plaincopy to clipboardprint
1. ClientContext context = new ClientContext("http://localhost");
2.
3. SP.List clients = context.Web.Lists.GetByTitle("Clients");
4.
5. SP.ListItemCollection myItems = clients.GetItems(new Caml
Query());
6.
7. //context.Load(clients);
8. context.Load(myItems);
9. context.ExecuteQuery();
10.
11. lst.ItemsSource = myItems;
12. lst.DisplayMemberPath = "{Title}";
```

ClientContext permet de se connecter au SharePoint distant, je récupère ensuite la liste concernée ainsi que ses items (remarquez que je passe une requête CAML vide, nous y reviendrons). La méthode Load sert à charger l'objet, puis ExecuteQuery envoie toutes les requêtes d'un coup au service SharePoint. Après l'accès aux éléments de la liste, passons au tri grâce à plusieurs moyens :

#### • Requête CAML

```
view plaincopy to clipboardprint
1. SP.ListItemCollection myItems = clients.GetItems(new Caml
Query())
2. {
3.     ViewXml = @"<View>
4.         <Query>
5.             <Where>
6.                 <Neg>
7.                     <FieldRef Name='Title' />
8.                     <Value Type='Text'>DOLLON</Value>
9.                 </Neg>
10.            </Where>
11.        </Query>
12.    </View>"
13. };
```

#### • LoadQuery & Mot clé Where

```
view plaincopy to clipboardprint
1. context.Load(myItems, items => items.Include(item => item
["Title"]).Where(item => (string)item["Title"] != "DOLLON"));
2. context.ExecuteQuery();
```

Cette requête affichera tous les éléments, exceptés ceux qui s'appellent "DOLLON". L'appel à la méthode Include permet de charger

l'index Title et la méthode Where réalise une comparaison. Nous aurions pu utiliser la méthode LoadQuery à la place de Load dans cet exemple. Pour finir, parlons de CRUD. Comment faire l'ajout, la modification et la suppression ? Et bien de cette manière simple (comme en SP2007) :

```
view plaincopy to clipboardprint
1. myItems[0]["Title"] = "Heliot";
2. myItems[0].Update();
3. context.ExecuteQuery();
```

## Les SharePoint REST Services

Pour communiquer avec SharePoint nous avons plusieurs possibilités comme vu précédemment. L'une d'entre elles consiste à utiliser l'architecture REST mise en place pour accéder aux données. Petit rappel sur REST, c'est un terme pour désigner une source de données accessible grâce au protocole HTTP de manière CRUD (Create, Read, Update, Delete). L'API Rest est basée sur WCF Data Services (aussi appelé Astoria 1.5), on peut donc accéder aux données au format ATOM ou JSON (donc compatible Silverlight & Ajax) de cette manière : [http://url\\_du\\_site/\\_vti\\_bin/ListData.svc/Nom\\_liste](http://url_du_site/_vti_bin/ListData.svc/Nom_liste)

Avec la possibilité que propose Astoria c'est-à-dire les filtres dans l'URL, le tri... On peut attaquer directement en ajoutant une référence dans Visual Studio au service listdata.svc.

## Les applications "sandboxées"

Ces applications sont exécutées dans une "sandbox" et sont déployées dans une gallery spécifique. Elles peuvent être de tous types comme Web parts, List templates ou encore workflow. La seule différence est que l'exécution du code implique une meilleure protection puisque la portée de l'application ne peut dépasser la collection de site. On se retrouve alors limité lors de l'utilisation de l'API (pas de SPSite par exemple). Vous pouvez choisir d'utiliser les applications sandboxées à la création d'un nouveau projet SharePoint : [Fig.28].

## Conclusion

Cette version est un vrai plaisir. Le développeur .NET, pas forcément compétent sur une plate-forme SharePoint, retrouvera très vite ses marques. Le seul petit bémol est l'utilisation de ressources que nécessite ce merveilleux outil ! Tous les développeurs n'ont pas une machine avec les performances nécessaires pour développer.

### ■ Julien Dollon

Most Valuable Professional / Consultant Trainer I'FORM/Exakis / Dotnet-France  
Community Leader / SUPINFO Full Professor .NET Technologies

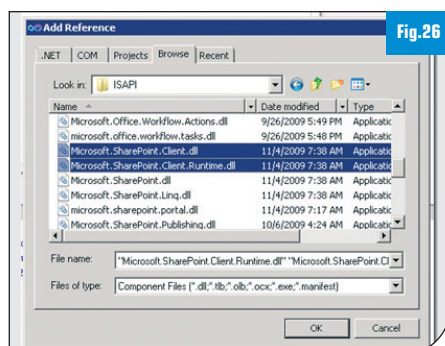


Fig.26

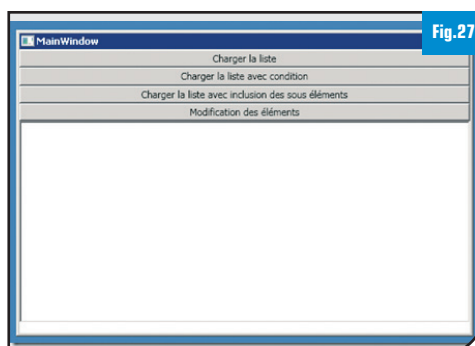


Fig.27

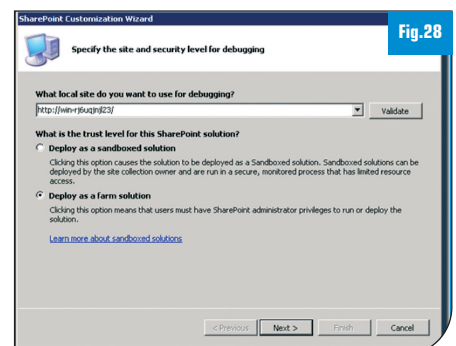


Fig.28

# Intel Energy Checker : votre logiciel est-il vert ?

Le kit de développement Intel Energy Checker propose un cadre, une API et des outils pour créer des métriques logicielles significatives pour l'efficacité énergétique des applications. Le code source est disponible sur [www.programmez.com](http://www.programmez.com)

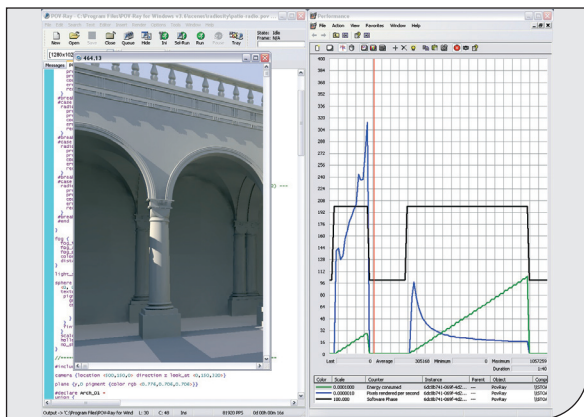
**U**ne fois passé l'effet d'aubaine du *greenwashing*, s'interroger sur l'efficacité énergétique des systèmes d'information est important à plus d'un titre.

Premièrement, et de façon très pragmatique, parce que nous payons l'énergie que nous consommons, et ce, à un prix sans cesse croissant.

Les administrateurs de fermes de calculs sont bien placés pour en témoigner.

Deuxièmement, parce qu'environ 2% de l'émission de dioxyde de carbone mondial est imputable aux systèmes d'information, ce qui correspond à plus de 28 trillions de tonnes de CO2 en 2006 selon les Nations Unies. Réduire cette quantité, ne serait-ce que d'un dixième de pourcent, serait incontestablement bénéfique pour tous.

*Impact de la nature des données sur la consommation d'énergie (courbe verte) – mais aussi sur la performance (courbe bleue). La courbe noire correspond à une variable d'état qui indique si le programme est oisif ou s'il calcule. Le convertisseur pl2w du kit est utilisé pour convertir le PL et ses compteurs en compteurs natifs Windows.*



## Travail et énergie

Nous avons conçu le kit de développement Intel Energy Checker justement dans le but d'aider les développeurs à être au fait du profil énergétique de leurs programmes, puis, in fine, à innover et améliorer l'efficacité énergétique de ces mêmes logiciels. Comme nous le détaillerons par la suite, plusieurs options s'offrent aux éditeurs en fonction de leur degré d'implication. Le kit s'adresse également aux utilisateurs au sens large. Il s'agira pour ces derniers de mesurer la consommation électrique de leurs systèmes pour mieux les gérer. Enfin, à la frontière des deux, les éditeurs de middleware pourront étendre les outils de contrôle qu'ils offrent aux administrateurs en y ajoutant l'efficacité énergétique. Mais qu'est ce donc que l'efficacité énergétique pour un logiciel ? Dans le cadre du kit de développement, nous la définissons comme la quantité de travail utile effectuée par le logiciel rapportée à la quantité d'énergie consommée par l'ensemble des équipements nécessaires à générer ce travail utile.

## Le programmeur sait mieux que quiconque

Nous avons longtemps cherché un proxy pour la quantité de travail utile effectuée par un code, sans vraiment y parvenir. En effet, prenons par

exemple la charge processeur qui est l'un des proxies le plus souvent cité. Appliquons-le à l'exemple trivial d'un programme souhaitant lire un message que Windows lui enverrait. S'il le fait avec PeekMessage() ou GetMessage(), sa charge processeur sera différente – ainsi que sa consommation, de par la nature bloquante du second. Pourtant, dans ce cas précis et fonctionnellement, ils font la même chose. Exit donc la charge processeur. En d'autres termes, ce n'est pas parce que vous passez des heures sans compter au bureau que votre travail sera efficace. Très rapidement, nous avons donc pris le parti de laisser les développeurs définir quel est le travail utile de leurs programmes. Cela a d'ailleurs du sens puisque par nature le logiciel est divers. Le travail sera ainsi l'exécution de requêtes SQL, d'encodage d'images, d'envoi de mégaoctets de courriels, ou bien encore le rendu de pixels. C'est pour cette raison que nous avons recours à l'instrumentation du code via l'API du kit de développement. En effet, l'unicité de chaque logiciel met en défaut des techniques transparentes comme l'instrumentalisation du binaire. Ainsi, l'action minimale requise de la part des éditeurs qui souhaitent utiliser le kit est de définir ce que leur programme fait d'utile et d'utiliser l'API pour rendre cette information accessible aux autres programmes – nous disons « exporter ». D'expérience, s'il est certain qu'un éditeur sait a priori quel est le travail utile réalisé par son logiciel, c'est la granularité du décompte qui peut être sujet à une discussion souvent passionnante. Aussi, ce qui est utile pour l'éditeur, ne l'est peut-être pas pour son client. Le kit est très flexible à ce niveau et toutes les combinaisons sont permises. Notons enfin, que la grande majorité des applications intègre déjà le code nécessaire pour compter la

## Watts ou joules ?

Pourquoi mesurer des joules plutôt que des Watts, qui sont la donnée la plus souvent communiquée lorsque l'on traite de la consommation électrique ? Premièrement parce que EDF facture des kWh et pas des Watts. Deuxièmement, parce que la puissance, mesurée en Watt donc, est le taux auquel un équipement consomme l'énergie. L'énergie est l'intégrale de la puissance dans le temps. Ainsi, un joule – l'unité SI de l'énergie – correspond à une consommation d'un Watt pendant une seconde.

Un kWh valant 3 600 000 joules. Concernant l'efficacité énergétique d'un logiciel, il nous semble plus naturel de s'interroger sur la quantité d'énergie requise pour effectuer une tâche donnée, plutôt que de connaître la puissance moyenne consommée pendant le temps nécessaire au traitement. Même en mesurant avec précision cette durée, le résultat du produit de la puissance moyenne par la durée ne serait qu'une mauvaise approximation de l'énergie consommée.



quantité de travail effectué, tout simplement parce qu'elles fournissent souvent des indications de performance aux utilisateurs.

## Simple comme un fichier

Dans la mesure où les développeurs sont trop souvent sollicités pour ajouter des fonctions à leurs programmes – et consacrer des ressources non négligeables à acquérir les connaissances et les nouvelles techniques requises –, nous avons voulu que l'API principale du kit soit élémentaire à utiliser, voire d'utilisation triviale. Dans la mesure où tous les programmeurs savent utiliser un fichier, nous avons calqué notre API sur celle des E/S. Ainsi, seulement cinq fonctions sont offertes : `pl_open()`, `pl_close()`, `pl_write()`, `pl_read()` et `pl_attach()`. Typiquement, un programme souhaitant exposer la quantité de travail qu'il effectue, commencera par faire un appel à `pl_open()`. Puis, à chaque fois qu'il le souhaite – à chaque centaine d'unité de travail accomplie, toutes les dix minutes, ou lorsque la première des deux conditions précédentes se vérifie, etc. – il appellera `pl_write()`. En fin d'exécution, `pl_close()` sera appelée. Le listing 1\* montre un tel exemple d'utilisation de l'API. À partir de cet instant, lorsque le programme esquissé ci-dessus s'exécute, un logiciel externe, un middleware ou l'un des outils livrés avec le kit, peut importer les compteurs exposés pour les enregistrer dans le temps, et surtout, pour les corrélérer avec la quantité d'énergie consommée. Par exemple, le kit intègre des outils d'interopérabilité avec Windows, ce qui permet d'importer ces compteurs dans Perfmon, Logman, VTune, etc. `pl_csv_logger`. Un échantillon du kit, peut être employé pour importer les compteurs de l'application et celui de l'énergie consom-

mée pour les sauvegarder dans un format compatible avec la plupart des tableurs. Les données énergétiques proviennent d'un élément du kit : le serveur d'énergie ou ESRV. Signalons aussi l'existence de TSRV qui est son homologue pour la mesure de la température et d'humidité relative. Ce sont les principaux facteurs susceptibles d'influer de façon tangible sur la consommation d'énergie. ESRV utilise l'API du kit pour exposer ses compteurs. Ainsi, l'énergie consommée depuis le démarrage du serveur – ou depuis sa dernière réinitialisation – la puissance instantanée, le facteur de puissance, le voltage, l'ampérage, etc. sont exposés via des compteurs. Le serveur expose également des compteurs intégraux pour ces valeurs ainsi que des minima et maxima. Dès lors, n'importe quel logiciel peut accéder à ces données grâce à un simple appel à `pl_read()`. Si jusqu'à présent nous n'avons considéré que le cas d'un logiciel se contentant d'exposer la quantité de travail qu'il effectue, en laissant le soin de la corrélérer avec la quantité d'énergie consommée, il apparaît clairement que ce même logiciel peut très simplement calculer, de son propre chef, son efficacité énergétique pendant son exécution. Il lui suffit pour cela d'importer le compteur « énergie » d'une instance locale d'ESRV [listing 2]\*. Par exemple, avec IBM et dans le cadre d'un concept démontré lors de l'IDF 2009 à San Francisco, nous avons instrumentalisé le serveur de bases de données DB2. En schématisant, ce dernier comptait le nombre des requêtes SQL exécutées et mesurait l'énergie consommée par le serveur sur lequel il s'exécutait (un serveur IBM System x équipé d'un capteur enfoui) pour calculer – et pour exposer à son tour – le nombre de transactions effectuées par joule

consommé et le nombre de joules consommés par transaction exécutée. Suivant le même principe, nous avons instrumentalisé le logiciel de rendu par lancer de rayons POV-Ray pour qu'il rende compte du nombre de pixels calculés par joule et inversement. Cette même version est capable d'importer le coût instantané du kWh – exposé par un module utilisant l'API – pour calculer le coût de revient de chaque image calculée. Ce programme démontre de façon patente l'impact non négligeable qu'a la nature des données sur la quantité d'énergie consommée.

## ESRV vous simplifie la vie

Reste que la mesure de l'énergie consommée requiert un matériel dédié. Il s'agira par exemple d'un analyseur de puissance externe, d'une sonde intégrée à l'alimentation de l'ordinateur, voire même d'un module logiciel qui estime la consommation de la plate-forme via un modèle plus ou moins précis. Ainsi, un même programme, instrumentalisé une seule fois pour être au fait de son efficacité énergétique, fournira des mesures de celle-ci très précisément dans un laboratoire en utilisant un analyseur externe – pour effectuer des benchmarks, optimiser des algorithmes, étudier

## Précision de la mesure

ESRV, le serveur d'énergie livré avec le kit de développement découple l'accès aux informations énergétiques de la gestion complexe des équipements de mesure. Mais cela permet aussi de pouvoir adapter facilement la précision des mesures de l'énergie consommée aux besoins des utilisateurs. En effet, la précision des données énergétiques dépend de l'équipement auquel ESRV est connecté. Par exemple, si l'analyseur utilisé dispose de la fonction d'intégration matérielle de l'énergie, alors la mesure sera extrêmement précise. Avec l'analyseur WT210 de Yokogawa par exemple, la puissance est mesurée avec une précision de 0.1%, puis est échantillonnée 50 000 fois par seconde. En comparaison, un capteur enfoui – mais intégré d'usine à l'ordinateur – n'offrira qu'une précision limitée et n'aura pas de surcroît d'intégration matérielle de l'énergie. ESRV effectuera alors une intégration logicielle à une fréquence de 1 Hz. Bien que moins précise, cette seconde solution sera bien adaptée économiquement pour un déploiement dans une ferme de serveurs, alors que la première solution sera préférée pour équiper un groupe de développeurs chargé de l'optimisation énergétique.

## Fiche technique

- Intel Energie Checker SDK
- Disponible pour Windows, Linux, Solaris 10 et MacOS X
- Langage de développement : C, interface JNI intégrée, classes exemple Java et C#
- Disponible en téléchargement gratuit à <http://software.intel.com/en-us/articles/intel-energy-checker-sdk/>
- Libre de droits
- Non Open Source, mais le code source de l'API est fourni ainsi que celui de nombreux exemples et outils du kit. Certains composants sont livrés au format binaire.

(\* ) Consultez [www.programmez.com](http://www.programmez.com)

l'impact de différents composants matériels, etc. – et offrira des mesures moins précises, mais sur le très long terme en utilisant des capteurs enfouis dans les blocs alimentations des serveurs sur lesquels il est déployé en production. Signalons à ce sujet que le kit est livré avec un module ESRV (esrv\_simulated\_device) qui simule la consommation d'un ordinateur pour permettre aux développeurs d'écrire leur code sans nécessiter l'achat d'un appareil de mesure sur tous les postes de développement. Lors de la phase de test et de déploiement, il suffira de connecter l'équipement de mesure approprié au système qui exécute ESRV (ce système pouvant être local ou déporté). Comme nous venons de le décrire, l'un des avantages d'ESRV est de découpler la tâche complexe de la mesure de la consommation de la donnée énergétique. Sachez que plusieurs équipements de mesure sont pris en charge par la première mouture du kit, et que de nombreux autres seront ajoutés lors des mises à jour. Un module spécial sera d'ailleurs prochainement disponible pour mesurer la consommation d'énergie par composants (DIMM, chipset, etc.). Ce module nécessitera toutefois l'installation de sondes de mesure à même les composants électroniques, ce qui le réservera à un petit groupe d'utilisateurs.

## Heuristique énergétique

Que faire de l'efficacité énergétique une fois mesurée ? C'est précisément là qu'intervient le vivier de compétences évoqué en préambule. Chaque développeur peut en effet faire la différence en inventant les mécanismes susceptibles de réduire la consommation énergétique de leurs applications. Pour illustrer notre propos, prenons le cas d'un moteur de bases de données au fait de son efficacité énergétique. Supposons que ce dernier s'exécute sur un serveur à huit voies le lundi matin à l'heure d'affluence dans les services. Son efficacité énergétique résulte du nombre des requêtes servies et de la quantité d'énergie consommée par le serveur – qui dans ce cas est loin d'être négligeable. Vers l'heure du déjeuner, le nombre des requêtes diminue et l'effi-



*Un analyseur P3 Kill-A-Watt a été modifié pour transmettre les données mesurées par ondes radio à un récepteur connecté à un PC. Un module ESRV ad hoc a été développé. Il est désormais possible d'analyser la consommation énergétique d'un équipement électrique sans fil !*

cacité énergétique du moteur de bases de données chute. Ce dernier détecte cette variation et demande au middleware la migration de son instance sur un serveur à deux voies. Avec un nombre de requêtes similaire, mais sur un serveur plus économique – mais aussi moins performant –, l'efficacité énergétique remonte, peut-être même à son niveau initial. En fin de journée, le serveur de bases de données demande la désactivation des mécanismes d'économies d'énergie au système d'exploitation – du serveur à huit voies sur lequel il a migré par anticipation du pic d'utilisation en se fondant sur son historique d'activité – afin de profiter de toute la puissance de la machine et d'optimiser son efficacité énergétique.

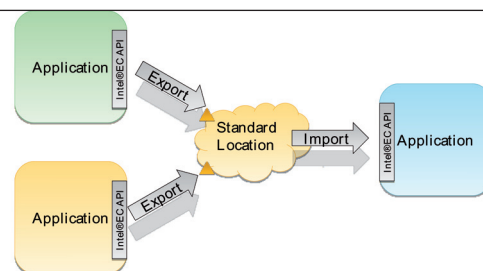
Prenons un second exemple. Imaginons que ce même serveur de bases de données traque sa qualité de service et l'expose – pour simplifier via un compteur d'état binaire : qualité de service assurée ou pas. Supposons que le middleware – mais cela pourrait très bien être le matériel lui-même – soit programmé pour activer agressivement des mécanismes de conservation d'énergie (par exemple en jouant sur le P-State). Ce middleware peut importer le compteur « qualité de service » du serveur de bases de données et arrêter le throttling. Sans ce mécanisme de coopération, soit le middleware, suivant un principe de précaution, laisserait des économies d'énergie sur la table, soit il ruinerait la qualité de service du logiciel de l'entreprise. Les possibilités sont littéralement infinies !

■ Jamel Tayeb

## Une question de vocabulaire

Dans le cadre du kit de développement, un compteur est une zone mémoire de huit octets qui reçoit un entier non signé de 64 bits. Comme son nom le laisse supposer, un compteur sert à compter le nombre de fois qu'un événement se produit, à l'instar d'un odomètre qui compte le nombre de kilomètres parcourus par un véhicule. Cependant, il est tout à fait possible d'utiliser un compteur comme une variable d'état. Par exemple sous une forme binaire, un logiciel peut utiliser un compteur pour indiquer s'il est capable ou non d'assurer sa qualité de service. Enfin, il est possible de représenter des nombres réels via une combinaison de compteurs dite suffixe.

Lorsqu'un programme souhaite mettre à jour un de ses compteurs (en appelant `pl_write()`), nous disons qu'il exporte le compteur. Une fois exportée, la nouvelle valeur est accessible (via un appel à `pl_read()`) par n'importe quelle application (qui doit faire un appel à `pl_attach()` au préalable). Nous disons qu'il importe le compteur. Les compteurs sont regroupés en PL. Un PL – Productivity Link – n'est rien d'autre qu'une organisation logique de compteurs. À chaque PL ouvert (par un appel à `pl_open()` ou `pl_attach()`) est associé un descripteur. C'est cet entier qui est le principal argument des fonctions de l'API. En guise d'organisation, il peut être utile par exemple de regrouper des compteurs associés à différentes fonctions du logiciel dans des PL distincts. Pareillement, il peut s'avérer être pratique de dédier un PL par processus léger d'un pool, chaque PL ayant – mais ce n'est pas une obligation – des compteurs identiques. L'accès à chaque compteur se fait par un déplacement dans son PL. Pour l'heure, les PL sont répliqués dans le système de gestion de fichiers, ce qui assure leur pérennité en cas de crash – un pré requis exigé par les opérateurs télécom. Enfin, chaque PL est associé à un GUID pour éviter des collisions de données une fois agrégées et chaque compteur à un nom. Le kit est agnostique en ce qui concerne la couche de transport utilisée choisie par l'utilisateur pour réaliser l'agrégation des PL dans un déploiement distribué. Rappelons enfin que l'utilisation du système de fichiers est un détail d'implémentation qui pourra changer dans le futur si nécessaire.



*Exemple d'une application important des compteurs exportés par deux autres applications.*



# egilia<sup>®</sup>

## LEARNING

LE SPÉCIALISTE DE LA  
**FORMATION CERTIFIANTE**  
EN **INFORMATIQUE**  
ET **MANAGEMENT**

Faire de vos succès  
notre réussite

[www.egilia.com](http://www.egilia.com)

CONTACTEZ NOS CONSEILLERS FORMATION

 **N°National 0 800 800 900**

APPEL GRATUIT DEPUIS UN POSTE FIXE

ANVERS . LIEGE . PARIS . LYON . LILLE . AIX-EN-PROVENCE .  
STRASBOURG . RENNES . BRUXELLES  
TOULOUSE . BORDEAUX . GENEVE . LAUSANNE . ZURICH .



# PHP a-t-il encore un avenir?



Le 11 mars 2010, Rasmus Lerdorf, le créateur de PHP, a officiellement déclaré l'arrêt du développement de PHP 6. PHP 5.3 semble n'avoir aucun successeur, alors que pratiquement 5 années ont été consacrées à son développement. Il est donc légitime de se demander si PHP a encore un avenir après ce qui peut sembler être un beau gâchis. Pour répondre à cette question, il est nécessaire de comprendre les raisons techniques et politiques qui ont mené à l'abandon de PHP 6.

## Les raisons de la crise

Techniquement, la solution choisie pour supporter Unicode au sein du langage s'est révélée être complexe à mettre en œuvre. Elle aurait également induit une dégradation significative des performances. De plus, la compatibilité de PHP 6 n'était pas garantie avec les versions antérieures, ce qui va à l'encontre de l'un des credo de développement du langage. D'un point de vue politique, le choix de développer PHP 6 parallèlement à PHP 5 a compliqué et augmenté la charge de travail des contributeurs. Les fonctionnalités ajoutées dans PHP 5 devaient en

effet être également intégrées dans PHP 6, en prenant en compte Unicode et la complexité qu'il apportait. De ce fait, une grosse majorité des contributeurs ont choisi de concentrer leurs efforts uniquement sur PHP 5 au détriment de PHP 6.

Les conséquences directes de ce désintérêt des contributeurs de PHP 6 ont été le très fort ralentissement de son développement et l'intégration dans PHP 5 de la plupart des fonctionnalités qu'il devait apporter. Il est alors logique de considérer que la dernière version de PHP 5 est un PHP 6 sans support natif de l'Unicode. Le développement de PHP 6 ne

se révèle donc pas être un gâchis complet. Toutefois, l'arrêt du développement de PHP 6 a eu un impact important sur la communauté.

## L'impact sur la politique de développement

La communauté des contributeurs a compris que la politique de développement de PHP 6 était trop contraignante et l'a donc complètement modifiée. La prochaine version de PHP est maintenant développée dans la continuité de la version actuelle, et non plus parallèlement. Son chantier a été mis en route rapidement afin de ne plus perdre de temps et permettre



aux contributeurs d'intégrer et tester leur code sans délai. De plus, son code de départ est celui de la dernière évolution de PHP 5.3 qui est connu et maîtrisé par l'ensemble de la communauté. Ensuite, le principe inauguré avec PHP 5.3 qui consiste à nommer deux gestionnaires de version a été reconduit. Ce tandem sera chargé de l'organisation et du suivi technique du développement de la prochaine version, dans le but d'améliorer la communication entre les développeurs et la qualité générale du code. Ils n'ont pas encore été élus par la communauté, mais il y a plusieurs candidats et une décision sera prise prochainement.

Afin de ne pas refaire un mauvais choix, toute modification du langage, que ce soit l'ajout ou le retrait d'une fonctionnalité, doit faire l'objet d'une "RFC" ("Request For Comment") débattue fonctionnellement et techniquement sur la liste de diffusion des contributeurs. À cette fin, l'ensemble des "RFC" est consultable publiquement sur le "wiki" de PHP.

Enfin, il y a une volonté de raccourcir les cycles de développement du langage pour ne pas perdre de temps si l'intégration d'une fonctionnalité pose un problème similaire à celui posé par l'intégration d'Unicode dans le cas de PHP 6 malgré l'utilisation des "RFC". La communauté a donc tiré les leçons de l'échec du développement de PHP 6. Elle s'est donnée les moyens de motiver à nouveau ses membres en simplifiant le processus de développement et en utilisant des solutions éprouvées qui limitent les risques de problèmes. Son action a d'ailleurs déjà porté ses fruits.

### Les effets du plan de relance

Depuis le début du développement de la nouvelle version il y a une soixantaine de jours, de nombreuses modifications et d'importants ajouts ont été effectués sur le code. Plusieurs "RFC" ont été approuvées et les fonctionnalités qu'elles décrivent implémentées. Parmi celles-ci, les plus significatives sont les "traits" qui viennent enrichir le modèle objet, le puissant outil de débogage DTrace, ou bien encore le

module CGI FPM, particulièrement adapté aux sites à fort trafic. Cela a été d'autant plus rapide que pour certaines, le code correspondant avait été développé dans le cadre de PHP 6 et n'a nécessité que peu d'adaptation pour être intégré dans la prochaine version.

A contrario, des fonctionnalités dépréciées ou considérées comme néfastes dans le cadre d'un développement et qui devaient disparaître avec PHP 6 ont été retirées. C'est le cas notamment du mode sécurisé activé par la directive "safe\_mode", de l'importation automatique des variables externes via "register\_globals" ou de la sécurisation des chaînes de caractères à l'aide des "magic quotes".

En parallèle, un travail important de correction et d'optimisation a été réalisé sur le moteur du langage. Les premiers tests effectués sur des projets en production montrent une amélioration significative des performances qui permettra au langage de répondre plus efficacement aux besoins des sites à forte audience. La qualité globale du code a de plus été améliorée à l'aide des tests unitaires qui ont été corrigés, mis à jour et enrichis.

Par ailleurs, des "RFC" ont été proposées en vue d'intégrer d'autres fonctionnalités que celle prévue à l'origine avec PHP 6, par exemple l'"auto-boxing", qui permettrait, si elle est intégrée dans le langage, de convertir de manière transparente un type natif en objet. Évidemment, il y a eu également énormément de bugs corrigés et des fonctionnalités améliorées, par exemple les "closures" dont les possibilités ont été étendues. Les contributeurs semblent donc motivés par le choc causé par l'abandon de PHP 6 et grâce aux décisions de la communauté.

### La dette de PHP 6 n'a pas encore été effacée complètement

Malgré tout les efforts faits jusqu'à maintenant, il reste des questions en suspens. La plus anecdotique concerne le nom que portera la prochaine version du langage. La communauté n'est pas parvenue à choisir entre PHP 5.4, PHP 6 ou bien toute autre

chose. En attendant qu'une décision soit prise, sa dénomination temporaire est 5.3.99-dev.

La question la plus symbolique concerne l'implémentation d'Unicode. Pour l'instant, il n'y a aucune discussion publique à ce propos dans la communauté et il n'existe aucune "RFC" proposant une solution technique. Il est difficile de dire si ce sujet fait peur du fait qu'il est la cause de l'abandon du développement de PHP 6 ou si les contributeurs ont besoin de temps pour trouver la solution permettant d'implémenter cette fonctionnalité sans sacrifier les performances. Il y a tout de même quelques contributeurs qui s'intéressent au problème, car quelques pistes ont été évoquées ponctuellement, comme étendre les possibilités des extensions "intl" ou "mbstring".

### Les utilisateurs et les professionnels croient en l'avenir du langage

Les utilisateurs du langage sont actifs. Il y a de plus en plus de publications, telles que des articles, des conférences ou des livres blancs, qui démontrent que l'industrialisation de projets basés sur PHP est possible et fonctionne parfaitement.

Dans le même temps, une bonne maîtrise de la programmation orientée objet ainsi que d'un ou plusieurs "framework" de qualité professionnelle est de plus en plus exigée dans les offres d'emploi relatives au langage. Ces mêmes "framework" sont par ailleurs en train d'évoluer afin de supporter les nouvelles fonctionnalités offertes par la dernière évolution de PHP 5.3, et plus particulièrement les espaces de noms. Ils pourront alors s'intégrer encore plus facilement dans un projet professionnel, ce qui facilitera encore plus leur adoption.

PHP se professionnalise donc de plus en plus et les concepteurs des "framework" à la mode ont suffisamment confiance en son avenir pour se lancer dans un travail d'adaptation important afin que leur projet supporte ses fonctionnalités les plus récentes.

Il est donc difficile de remettre en question l'avenir du langage.



## La signature de l'avis de décès semble prématurée

Les évolutions successives de PHP 5 ont apporté aux utilisateurs la majorité des fonctionnalités qui devaient être supportées par PHP 6. L'arrêt du développement de ce dernier signifie uniquement le report du support natif d'Unicode qui se fera techniquement d'une autre façon que celle prévue à l'origine. Par ailleurs, si la communauté a tardé à prendre conscience des problèmes dans le développement de PHP 6, elle a cependant réagi rapidement et efficacement. Elle a supprimé tout ce qui avait démotivé les contributeurs et elle a appliqué des mesures correctrices issues de l'expérience acquise lors du développement de PHP 5.3. En conséquence, les contributeurs ont repris plaisir à travailler sur le langage et le développement de la prochaine version progresse rapidement. Il y a bien des problèmes qui n'ont pas encore de solutions, mais les développeurs ont des pistes pour les traiter et les résoudre n'est qu'une question de temps. Les professionnels sont de plus en plus demandeurs de compétences pointues sur le langage et ses utilisateurs croient en son avenir. La communauté française, représentée par l'AFUP, n'est d'ailleurs pas la dernière à y croire puisqu'elle organise le Forum PHP qui se déroulera à la Cité des Sciences de Paris la Villette les 9 et 10 novembre 2010. PHP n'est donc pas mort avec l'abandon du développement de PHP 6. Il est même possible de dire que toutes les conditions sont réunies, tant au niveau des contributeurs que des utilisateurs, pour que PHP continue à vivre encore longtemps.

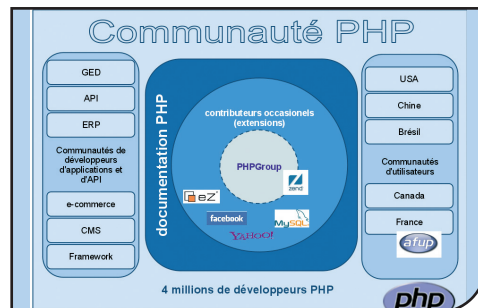


■ **Frédéric Hardy** utilise PHP professionnellement depuis plus de 10 ans. Architecte d'application, administrateur système et infographiste ergonomiste, il est l'auteur du blog

<http://blog.mageekblog.net/>. Il aime y parler de technique en général, avec une affection particulière pour FreeBSD et PHP.

# La communauté PHP : son organisation

Dans cet article nous allons tenter de vous présenter le fonctionnement, parfois un peu obscur, de la communauté PHP.



Il y a presque quinze ans Rasmus Lerdorf créait PHP en codant un ensemble de scripts Perl, dont il se servait pour noter les accès à son CV en ligne. Pour ceux qui se souviennent, à cette époque, le Web n'en était qu'à ses balbutiements et les rares applications dynamiques étaient des scripts cgi souvent codés avec Perl. Depuis, PHP a bien évolué que ce soit en termes de code, de fonctionnalités mais également d'écosystème.

Le cœur de la communauté PHP est dirigé par ce que l'on appelle le PHPGroup. C'est l'organe le plus puissant dans la mesure où ce sont ses membres qui décident de la direction que prend PHP. Sa structure est assez floue et mouvante mais on peut consulter sur la page des crédits de PHP une liste où sont indiqués ses principaux acteurs (<http://php.net/credits.php>). Note importante, le PHPGroup est composé uniquement de profils techniques ; les problématiques marketing et commerciales n'entrent pas en compte. De nombreuses voix s'élèvent régulièrement sur la façon un peu autoritaire dont est dirigé PHP. En effet il n'est pas rare de voir des patches complets proposés à la revue qui ne sont pas exploités tout simplement parce que leur auteur n'est pas renommé dans le monde PHP.

Autour du PHPGroup, qui dirige le développement du cœur de PHP, il y a les contributeurs plus occasionnels ou ponctuels. Ils travaillent sur ce qu'on appelle les extensions de PHP. Pour comprendre, il faut savoir que PHP a été créé de façon modulaire : son cœur est une sorte de contrôleur sur lequel on

peut ajouter des fonctionnalités spécifiques via des extensions. Par exemple, pour générer des images, on installe l'extension GD.

Dans ce cas, ce sont souvent des développeurs spécialisés qui interviennent sur ce spectre.

Lors d'un forum PHP organisé par l'AFUP, Rasmus Lerdorf avait indiqué que la première version du connecteur entre PHP et Oracle avait été développée rapidement par le PHPGroup pour initier le mouvement. Avec un connecteur, même peu performant, les entreprises ont commencé à l'utiliser et, parfois, pour se heurter à certaines limites. C'est là qu'entrent en scène les entreprises contributrices.

Plusieurs sociétés contribuent régulièrement. Leurs intérêts sont divers. Les plus connues sont Yahoo!, MySQL, eZ, Zend et FaceBook. Pour Yahoo! le leitmotiv est "performances". Cela se comprend aisément quand on sait que 99% du frontal Web de Yahoo! est écrit en PHP. Pour être sûr d'avoir accès au cœur de l'information, ils ont employé pendant quelques années Rasmus Lerdorf (lequel a annoncé qu'il était temps de changer quand Yahoo! et Microsoft ont annoncé un partenariat - il ne fait plus partie de Yahoo!). Pour MySQL le lien a été historiquement plus fort, les deux ont profité de l'aura de l'autre pour avancer et se faire reconnaître. Enfin, la société Zend qui a toujours eu une relation spéciale avec PHP : d'une part parce que le moteur de PHP est développé par eux, et d'autre part parce que la nature et les ambitions commerciales de



la société ont toujours été vues d'un mauvais œil. De nombreux développeurs de PHP reprochent, en effet, à Zend de trop tirer la couverture à eux. Avec le temps, un statu quo semble s'être installé. Une des grandes forces de PHP réside dans sa facilité d'apprentissage et sa documentation.

Il s'agit de la troisième couche de la communauté. Un système complet de documentation a été mis en place et des équipes travaillent régulièrement pour traduire la documentation et la maintenir à jour. Les quasi anonymes qui travaillent régulièrement sur ces aspects sont des contributeurs essentiels au développement de PHP. Documentation de PHP : <http://www.php.net/docs.php>

Un autre pan de la communauté PHP est la communauté d'utilisateurs composée de plus de 4 millions de développeurs. Énormément de pays disposent d'une antenne locale. Ces groupes d'utilisateurs sont généralement actifs et organisent de façon plus ou moins large la vie autour de PHP dans leur pays. En France c'est l'AFUP, Association Française des

Utilisateurs de PHP qui est, de facto, le représentant de PHP pour les particuliers et les entreprises.

AFUP : <http://afup.org/>



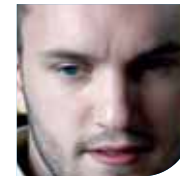
Depuis dix ans l'AFUP fait vivre PHP en France en organisant des séminaires, des retours d'expérience, en publiant des opinions dans la presse et, point d'orgue de l'année, en organisant le Forum PHP (9 et 10 novembre 2010 à La Cité des Sciences et de l'Industrie à Paris). Si vous êtes un professionnel utilisant PHP, vous êtes invité à rejoindre les rangs de cette communauté active ! En France les principaux sites pour apprendre PHP sont PHPFrance.com, PHPDebutant.org, siteduzero, développez...

Quid du futur ? PHP est stable et fiable, son évolution est maintenant à son rythme de croisière. Les nouveautés ne se situent plus sur la plateforme PHP même mais sur les outils fonctionnant dessus. De nombreuses entreprises et des particuliers ont besoin d'un outil de gestion de contenu, de nombreuses

entreprises ont besoin d'une boutique en ligne, de générer des fichiers PDF, OpenOffice, MS Office, ... Actuellement, c'est là que se situe l'innovation. Les nombreuses offres d'emplois ne trouvant pas preneur sur Drupal, Symfony, Magento, Prestashop et ZFramework témoignent de la bonne forme de l'écosystème PHP.



**■ Cyril Pierre de Geyer**  
Co-auteur du livre « PHP 5 avancé » et vice-président de l'AFUP, Association Française des utilisateurs de PHP, il participe aux sites PHPteam.net et PHPFrance. Impliqué dans l'écosystème Open Source. Son expertise l'a amené à intervenir lors de nombreuses conférences nationales et internationales.



**■ Nicolas Silberman**  
Après 20minutes.fr, Nicolas Silberman est responsable technique des éditions digitales du groupe Condé Nast (Vogue, Glamour, GQ, ...). Il est également depuis début 2010 le Président de l'AFUP, Association Française des Utilisateurs de PHP

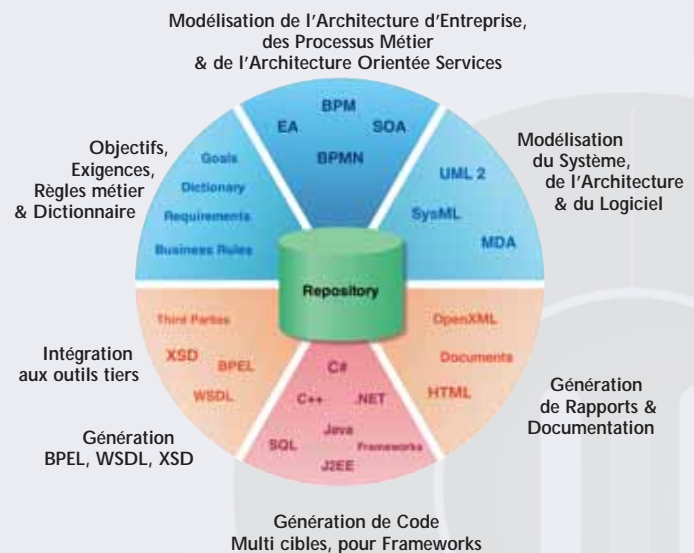
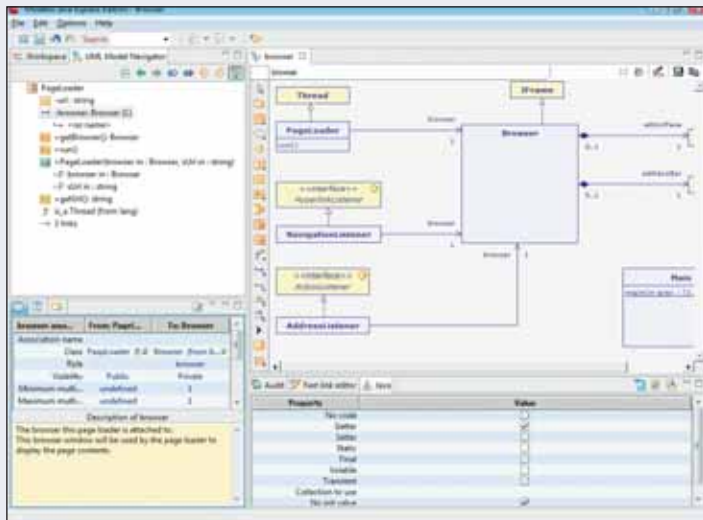


## Modelio : une offre de modélisation unique

*Libérez la vraie puissance de vos modèles !*

### UML, BPMN, Exigences ..., MDA, génération de code ...

- Modélisation intégrée de UML2, BPMN, SysML, l'Architecture d'Entreprise, les exigences, le dictionnaire, ... dans un seul référentiel
- Génération Java, C#, C++, SQL, XML, XSD, BPEL, WSDL, Hibernate...
- MDA simple et puissant - transformation, extensibilité et adaptabilité
- Travail de groupe distribué, intégré à SVN/Subversion
- Ergonomie simple, productive et familière aux développeurs (RCP/Eclipse)



### Modelio est disponible en trois éditions

- **Free** : Un outil de modélisation UML2, BPMN, et de développement MDA, complet et gratuit !
- **Express Java** : Un outil de développement UML2/Java performant pour seulement 100 € !
- **Enterprise** : La solution de modélisation complète, supportant le travail de groupe, extensible avec une riche palette de modules de modélisation et de génération disponibles sur étagère



Téléchargez la nouvelle version de Modelio !  
[www.modeliosoft.com](http://www.modeliosoft.com)

[sales@modeliosoft.com](mailto:sales@modeliosoft.com)  
Tél. : 01 30 12 18 40



# PHP : choisir un bon outil

Lorsque vous choisissez une technologie ou un langage, le premier réflexe consiste à rechercher des outils adaptés à vos besoins. Mais si vous avez déjà un outil qui supporte PHP et les technologies web, peut-être n'est-il pas la peine d'en changer, sauf à trouver l'oiseau rare.

L'apprentissage d'un outil est à l'image d'un langage, vous devez apprendre à vous en servir, à le maîtriser. Cet outil se doit d'être simple et facile à utiliser pour éviter de passer de nombreuses heures d'apprentissage en plus du langage proprement dit. L'utilisateur d'un IDE sera souvent précieux, car il agrège de nombreux modules utiles au développement et évite de jongler entre plusieurs outils non intégrés.

## Un environnement

Lorsque vous souhaitez passer au langage PHP, vous devez installer et utiliser un environnement de développement, de déploiement et d'exécution pour vos pages web créées en PHP. On pourra utiliser du très classique : Apache, PHP, MySQL.

La majorité des hébergeurs gratuits et professionnels proposent cette configuration. Mais vous pouvez posséder votre propre environnement sur votre machine ou mini-serveur. Vous trouverez par exemple Wampserver sous Windows, Mamp sous Mac, Xampp sous Linux. Bien entendu, le langage fonctionne sous d'autres configurations, avec la possibilité de réaliser votre propre environnement.

## L'outil le plus simple

Le langage PHP est un langage de script offrant différentes possibilités de programmer comme le mode procédural ou le mode objet. Il se présente comme ceci :

```
<?php
echo « Hello world ! »;
?>
```

Il affichera sous votre navigateur, le message Hello world ! Pour réaliser cette opération, vous pouvez utiliser directement un bloc-notes ou un éditeur Basic qui est fourni par votre système d'exploitation (Linux, Mac, Windows). Il est très facile d'apprendre à utiliser l'outil, mais vous serez

confronté à de nombreux problèmes qui vont apparaître rapidement comme la détection des bugs, des tests, des livraisons vers le serveur...

## Un outil évolué

Un outil évolué permet de bénéficier des fonctionnalités que peut attendre un développeur. Certains outils sont utilisables sur une configuration précise, d'autres se retrouvent sur une multitude de systèmes. L'approche multiplateforme est un critère important si vous devez utiliser des environnements différents, tout en gardant le même modèle de développement et d'outillage. Cela évite d'apprendre plusieurs outils. Bien entendu, l'éditeur de texte ou de code doit pouvoir vous permettre de modifier les scripts en PHP. Vous n'aurez pas besoin de rechercher un compilateur car avec le langage PHP, il n'est pas nécessaire de compiler les sources pour voir un script fonctionner. PHP est avant tout un langage interprété.

Un débogueur vous sera utile pour faire fonctionner correctement votre projet web, il a pour rôle de repérer les erreurs éventuelles de votre programmation. Les erreurs classiques qui peuvent bloquer votre projet pendant de longues heures, sont souvent associées comme :

- Absence de point-virgule à la fin d'une ligne
- Absence d'une accolade dans une boucle
- Erreur de frappe dans l'utilisation d'une fonction du langage
- Absence d'une des 2 quotes dans une utilisation de variables
- Etc.

L'ensemble de ces fonctions et critères sont importants pour une bonne utilisation d'un langage comme PHP. Ces critères sont applicables à d'autres langages.

## Un outil web : pourquoi pas

Depuis quelque temps, les outils sur le web se multiplient, même s'ils restent margi-

naux dans leur usage. Les outils disponibles directement sur le web permettent de développer des projets web sans avoir votre poste de travail. Le développement s'effectuera par l'intermédiaire de votre navigateur à partir d'un accès sécurisé. La seule obligation consiste à posséder une connexion internet.

Vous trouverez actuellement comme « Web-based » : Ecco, phpAnywhere, codeRun... Les langages supportés sont nombreux mais si vous devez effectuer une compilation, vous ne pourrez pas le réaliser. Les contraintes de compilation sont encore trop fortes. Le plus connu et un des premiers à apparaître sur le marché est « phpAnywhere ».

## Choisir son EDI : le dilemme

Maintenant que les critères à prendre en compte sont connus, vous devez :

- Tester les différents outils
- Consulter les fiches techniques
- Définir les environnements
- Le budget

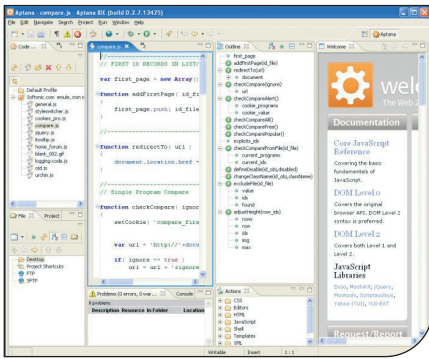
La plupart des outils présentés ici supportent plusieurs langages. Vous pourrez ainsi passer d'un langage à un autre quand vous le souhaiterez. C'est aussi un gage de pérennité de votre environnement de travail. Cependant, impossible de faire la liste de tous les outils disponibles sur le marché. Nous avons volontairement réduit la liste à quelques outils connus (gratuits ou payants). Cela vous permettra de vous faire une première idée.

### Webscripter

Il fonctionne exclusivement sur MacOS X et propose une utilisation assez simple avec de nombreuses fonctions comme l'accès FTP, la colorisation syntaxique, le débogage, etc. Son développement est suspendu et est disponible gratuitement.

<http://www.web-scripter.com/>





### Aptana IDE

Une des outils les plus connus et les plus utilisés car fonctionnant sur Windows, Mac et Linux ! Il possède une grande richesse fonctionnelle. Un système de plugins permet d'ajouter de nombreuses extensions. Une aide à la saisie est intégrée (très pratique) pour l'ensemble des langages que vous utilisez autour de PHP (html, css, javascript...). Disponible sous licence GPL v3/APL. <http://www.apptana.org/>

### NetBeans

NetBeans est un IDE multiplateforme sous licence Open Source et compatible pour tous les systèmes d'exploitation qui supporte tous les langages Web dont PHP, Ruby, Python. Un petit paramétrage est nécessaire pour définir le langage utilisé pour répondre aux différentes syntaxes des langages et des souhaits des programmeurs. <http://netbeans.org/>

### Eclipse PDT

Sous licence libre, Eclipse PDT est une distribution spécifique du projet Eclipse IDE. Il propose un environnement multilingue pour tous les types d'ordinateurs. Il couvre toutes les attentes des développeurs, et si une option n'est pas présente, vous pouvez l'installer avec la gestion de plugins intégrés.

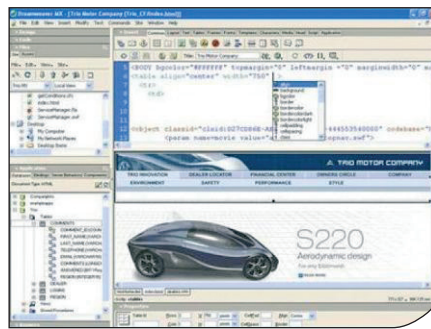
grés quel que soit votre environnement de développement (Linux, Mac, Windows). Cet éditeur basé sur Java, a été conçu avec une architecture de plugins. Le concept a été repris par d'autres EDI car l'avantage permet d'utiliser juste les ressources nécessaires à votre projet.

<http://www.eclipse.org/pdt/>

### Visual Studio Express

Avec la nouvelle version de ce logiciel (en anglais), vous pouvez facilement utiliser le langage PHP. Fonctionnant uniquement sous Windows, il s'agit d'une édition gratuite et allégée de Visual Studio. IDE assez complet, il permet de développer de nombreux projets ne nécessitant pas des modules de tests avancés mais il propose la colorisation du code, une documentation complète, des fonctions de debug, etc.

<http://www.microsoft.com/express/Web/>



### Dreamweaver

L'outil de création des sites web par excellence désormais disponible en version CS5 depuis quelques semaines.

Très complet, il supporte de nombreux langages web et possède une richesse fonctionnelle impressionnante pour le développeur web et notamment PHP. Disponible sous Windows et MacOS X. Payant. [www.adobe.com](http://www.adobe.com)

### Zend Studio

Cet EDI est considéré comme un outil de nouvelle génération et permet d'optimiser les applications PHP professionnelles et aussi l'utilisation du Framework du même nom. Il permet d'augmenter la productivité des développeurs et de maintenir le code plus facilement. Par ailleurs, l'outil de débogage a été optimisé pour résoudre les problèmes plus rapidement et améliorer le travail en équipe. Il est distribué sous licence commerciale et fonctionne sous Linux, Windows, Mac [http://www.zend.com/zend\\_studio](http://www.zend.com/zend_studio)

### PHPEdit

Il s'agit d'un des tout premiers EDI orienté PHP. Au fil des années, il a su évoluer pour proposer un outil de nouvelle génération avec une optimisation Framework PHP (Symfony, Prado, eZ publish). Il répond à l'ensemble des attentes des développeurs et est distribué sous licence commerciale pour une utilisation professionnelle et fonctionne sous Windows (prochainement sur MacOS). <http://www.phpedit.com/fr>

### PhpAnywhere

Cet outil évolue régulièrement et les fonctions essentielles de développement comme la colorisation du code, un explorateur de contenu en mode FTP, la fonction annulation, l'auto-complétion du code, etc. Il va vous permettre de modifier, de créer les pages web et se trouve compatible avec tous les langages de scripts qui ne nécessitent pas une compilation. Il fonctionne sous tous les systèmes d'exploitation. <http://phpanywhere.net/>

### ■ Christophe Villeneuve

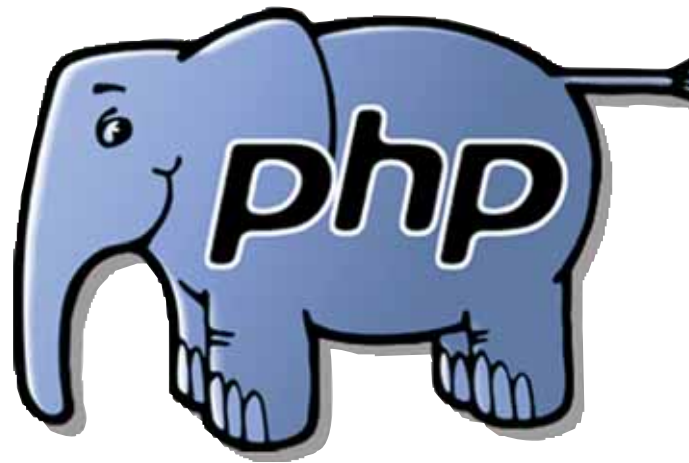
Consultant pour Alter Way solutions et auteur du livre « PHP & MySQL-MySQLi-PDO, Construisez votre application » 2e édition, aux Éditions ENI. Rédacteur pour nexen.net, Trésorier AFUP, LeMug.fr, PHPTV, PHPteam...

## Tableau récapitulatif

	Basic (Editeur de texte)	Webscripiter	Aptana	Netbeans	Eclipse PDT	Visual studio	Dreamweaver	Zend studio	PHP Edit	php Anywhere
Distribution	Basic avec votre ordinateur	Gratuit	GPL/APL	Gratuit	Gratuit	Payant	Payant	Payant	Payant	Web-based
Environnement	Linux Mac Windows	Mac	Linux Mac Windows	Linux Mac Windows	Linux Mac Windows	Windows	Mac Windows	Linux Mac Windows	Windows	Linux Mac Windows
Editeur de texte	•	•	•	•	•	•	•	•	•	•
Colorisation syntaxique	-	•	•	•	•	•	•	•	•	•
débogueur	-	•	•	•	•	•	•	•	•	-
Outils intégrés	-	•	•	•	•	•	•	•	•	•
Différents langages supportés	•	•	•	•	•	•	•	-	-	•

# Un éléphant aux pieds d'argile ?

Après 15 ans d'une évolution au rythme changeant, PHP a incontestablement gagné sa place parmi les plateformes de développement web majeures. Qu'il s'agisse de sites web ambitieux ou d'applications cœur-métiers, si un projet doit être réalisé avec des technologies web, il peut l'être avec PHP. La question qui doit se poser aujourd'hui n'est donc pas de savoir si PHP est mort, mais ce qui pourrait le tuer !



**A**vant de traiter de cette question, il me semble important de donner quelques informations complémentaires sur mes propres opinions sur le sujet, afin d'écarter toute suspicion de « partisanisme ». Professionnel du marché PHP et évangéliste autoproclamé de la technologie depuis plus de 10 ans, je suis convaincu des qualités intrinsèques de ce langage et de son écosystème, parmi lesquelles, et sans doute loin devant les autres, sa capacité à offrir des stratégies pour répondre avec une efficacité incomparable à la quasi-totalité des problématiques soulevées par le développement web, dont la masse et l'importance ne cesse de croître année après année. Aussi il est entendu que poser une telle question, de mon point de vue, ne constitue pas une accusation, mais représente une inquiétude, dont je me propose de partager les causes avec vous.

## Que de chemin parcouru...

Ainsi que cela a été rappelé dans l'introduction au présent article, PHP est désormais pris au sérieux par les entreprises, y compris pour des projets allant au-delà du petit site d'e-commerce. Ceci étant naturellement une façon imagée de rappeler combien PHP a été méprisé pendant de

nombreuses années. Bien sûr, des projets de grande envergure ont de tout temps été menés en PHP. Parfois même en dépit du bon sens, la technologie ayant mis un certain temps, force est de le reconnaître, pour atteindre le degré de maturité qui est le sien aujourd'hui. Mais ceci est caractéristique des technologies iconoclastes, portées par des aficionados très enthousiastes. Et ces exemples de réussites inattendues ont fait évoluer les mentalités au fur et à mesure que la plateforme elle-même évoluait.

Pour revenir à la situation actuelle, et la résumer, disons simplement que les décideurs considèrent désormais l'alternative PHP d'un point de vue purement objectif, et non par attachement personnel.

Des cabinets d'analyse aussi sérieux et reconnus que Gartner [1] se sont penchés sur le phénomène. Une note datant de décembre 2009 laisse entendre que le nombre de développeurs PHP va continuer de progresser à un rythme certes faiblissant mais soutenu dans les années à venir, pour atteindre 5 millions dans le monde entier en 2013 (contre 4 millions aujourd'hui). D'autres études, comme celle effectuée il y a quelques années par Globalis [2], nous indi-

quent que l'écrasante majorité des plus grosses sociétés française ont recours à PHP pour tout ou partie de leurs sites web.

Tout ceci devrait donc augurer d'un avenir radieux... et pourtant, je pense qu'un risque existe que cet écosystème s'affaisse sous le poids de son propre succès...

## Ce qui pourrait tuer PHP

La légitimité que PHP a obtenue en tant que plateforme majeure est un essai qu'il reste à transformer. Ce nouveau statut sur le marché permet à PHP de se voir confier des projets de plus en plus importants. Leur réalisation réclame donc des niveaux de compétence et d'expérience eux aussi revus à la hausse.

Si la population de développeurs PHP est très imposante, comme nous venons de le rappeler, l'expérience nous amène à relativiser cette donnée : le degré de savoir-faire constaté est trop souvent très en deçà des nouveaux besoins suscités tant par l'évolution technique du langage (l'orientation objet notamment), l'adoption de nouvelles méthodologies (design patterns, intégration continue...) que la criticité des projets auxquels les développeurs sont maintenant confrontés.

Je voudrais évoquer ici certains des

pièges dans lesquels la culture très "web" de PHP amène trop souvent les développeurs à tomber, culture qui peut se résumer comme suit "seul le résultat compte" :

## Pratiquer le "quick and dirty"

Cette attitude qui consiste à se concentrer sur la conséquence plutôt que sur la cause, dans le but de produire une solution immédiate, conduit inexorablement à la génération de nouveaux problèmes, le plus souvent sensiblement plus complexes que celui initialement résolu.

Appliquer une solution (qui en soit vraiment une) à un problème doit s'inscrire dans une démarche plus vaste, prenant en compte toutes les répercussions possibles, ce qui se fait en évaluant tous les scénarios possibles et en les testant de manière exhaustive et donc automatisée. Autrement dit, en respectant toutes les étapes habituelles du cycle de développement.

*“ ce qui est “dirty” n’est jamais “quick” (sur le long terme) ”*

## Se passer de frameworks

Considérer que partir de zéro dans le but de maîtriser la totalité d'un projet est, dans la plupart des situations, aussi illusoire que d'espérer trouver un outil qui fera tout à votre place. Ces dernières années ont vu apparaître de nombreux frameworks MVC (3) offrant une très grande richesse fonctionnelle et permettant de donner au développement une structure standardisée (et éprouvée), particulièrement adaptée au web. Entre autres (énormes) avantages, l'emploi d'un framework permet de renforcer rapidement une équipe, pourvu que le framework en question soit suffisamment répandu bien sûr, comme Symfony ou Zend Framework pour citer les deux plus couramment

rencontrés en France. Mais la réalisation d'un site ou d'une application en PHP implique en outre des technologies connexes, telles HTML, JavaScript ou CSS pour ne citer que les plus évidentes. Pour les prendre en charge, il existe également des frameworks et bibliothèques de grande qualité et dont il n'est pas question de se priver, comme jQuery, ExtJS ou encore Dojo, qui pour la plupart prennent en charge les trois, ainsi bien sûr que les échanges Ajax.

*“ si “fait maison” est la réponse, vous êtes très certainement en train de vous poser la mauvaise question (extrapolé d’une citation de Rasmus Lerdorf incriminant l’emploi d’eval()) ”*

## Ne pas s'ouvrir aux autres technologies

La généralisation des web services permet une interopérabilité très importante avec des systèmes issus d'univers très différents.

Un exemple typique est le serveur de recherche Solr, de la fondation Apache, qui offre des fonctionnalités d'indexation et de recherche très avancées. Solr est écrit en Java, mais est d'une simplicité confondante à interfacer en PHP.

Si cet exemple est très représentatif, c'est tout simplement parce que le service apporté par Solr est proprement inégalable en pur PHP.

Aussi, sans connaissance de l'existence de Solr, on se retrouve à créer manuellement, ce qui est long et difficile, un moteur de recherche en PHP qui offrira des performances calamiteuses et une pertinence toute relative.

*“ ce qui ne peut pas être fait en PHP ne doit pas être fait en PHP ! ”*

Il y a bien entendu de nombreuses autres bonnes pratiques à observer pour garantir la réussite d'un projet en PHP. Il s'agissait ici simplement de souligner un certain nombre de mauvaises pratiques qui ont encore largement cours et qui sont incompatibles avec les défis à venir.

## Tout ceci est-il inexorable ?

Naturellement non ! Certes le constat est un peu dur, et vient compléter l'intitulé plutôt provoquant du dossier dans lequel il s'inscrit. Mais non, cent fois non, la partie n'est pas perdue.

La conscience de ces problématiques est bien présente dans la communauté PHP, à l'image de l'évolution du langage lui-même qui tend à s'affranchir de ses défauts de jeunesse, qui pour certains auront perduré longtemps car considérés comme qualité ! Mais qu'importe s'il reste des fervents de la permissivité absolue des débuts de PHP, celle-ci se restreint, et continuera de se restreindre à mesure que les développeurs aspireront au respect de meilleures pratiques.

D'autre part, la qualité et la vitalité des frameworks sont vraiment le signe le plus évident d'une recherche de plus en plus répandue de standardisation des méthodes, de rationalisation des développements, en un mot d'industrialisation du cycle de développement en PHP.

## Conclusion

Même avec PHP, le développement est un métier ! Qu'il soit facile d'apprendre le langage ne doit pas dispenser d'apprendre les méthodologies et les bonnes pratiques qui seules lui permettront de pérenniser sa position durement acquise au sein d'un marché très concurrentiel, et terriblement exigeant.

■ Gauthier Delamarre

(1) "PHP: Past, Present and Future" ([http://blogs.gartner.com/mark\\_driver/2009/12/03/php-past-present-and-future/](http://blogs.gartner.com/mark_driver/2009/12/03/php-past-present-and-future/))

(2) "Place de la plate-forme PHP dans l'économie française" ([http://globalis.fr/IMG/pdf/PHPEnFrance\\_V2.1.pdf](http://globalis.fr/IMG/pdf/PHPEnFrance_V2.1.pdf))

(3) Model-View-Controller (<http://fr.wikipedia.org/wiki/Modèle-Vue-Contrôleur>)





# Unicode et les conséquences sur PHP 6

En novembre 2005, les développeurs de PHP se rencontrèrent à Paris pour discuter des évolutions de la sixième version du langage. Cinq ans plus tard, PHP 6 n'est toujours pas disponible. Examinons ensemble les causes de ce retard.

Les premières annonces de PHP 6 étaient très prometteuses :

- Ajout des namespaces : possibilité de lier des classes ou des fonctions à un « espace de nom ». Ces classes seront disponibles uniquement si on importe explicitement le namespace (voir le mot-clé « import » en Java ou Python) ;
- Static binding : permet d'effectuer la résolution des attributs statiques par la classe appelante et non par la classe mère ;
- Ajout des fonctions anonymes ;
- Du nettoyage : suppression des « register globals », « magic quotes » et « safe mode » ;
- Intégration d'un « goto » ;
- Intégration d'un cache de bytecode ;
- Et surtout l'Unicode.

## Unicode en PHP.

L'ambition de PHP est d'intégrer l'Unicode au cœur du langage. Ceci doit permettre l'utilisation de n'importe quel caractère dans le code source (oui, vous pourrez mettre des accents dans vos noms de variables, mais ne vous sentez pas obligé de le faire !). Mais le support de l'Unicode devrait surtout permettre d'utiliser des sources de données qui contiennent autre chose que de l'ASCII, de manière native, sans utiliser de bibliothèque externe (adieu mbstring). Les caractères de la langue anglaise ne seraient qu'un sous-ensemble du jeu de caractères supporté, et non pas une langue privilégiée. Voici un exemple qui met en évidence le fait que PHP ne gère pas l'Unicode. Prenons un fichier XML dont l'encodage est définie explicitement comme UTF-8.

```
$ cat exemple.xml
<?xml version="1.0" encoding="UTF-8"?>
<example>
  <string>_____ unicode</string>
</example>
```

Voici un parseur XML trivial (il n'y a aucune vérification d'erreurs). Le parseur utilise SimpleXML.

```
<?php
$xml = simplexml_load_file("exemple.xml");
$results = $xml->xpath('/example/string');
$str = $results[0];

echo "R_sultat simpleXML : ".$str;
echo "ÃnÃn";

echo "API de base:Ãn";
echo "taille de la chaine : ".strlen($str);
echo "Ãn";
```

```
for($i=0; $i < strlen($str); $i++) {
    echo substr($str, $i, 1);
    echo " ";
}

echo "ÃnÃnAPI unicode:Ãn";
echo "taille de la chaine : ".mb_strlen($str, "utf-8");
echo "Ãn";

for($i=0; $i < mb_strlen($str, "utf-8"); $i++) {
    echo mb_substr($str, $i, 1, "utf-8");
    echo " ";
}

echo "Ãn";
```

Et la sortie :

```
R_sultat simpleXML : _____ unicode

API de base:
taille de la chaine : 21
_ _ _ _ _ u n i c o d e

API unicode:
taille de la chaine : 14
_ _ _ _ _ u n i c o d e
```

On voit que les fonctions de base de PHP considèrent que tous les caractères sont codés sur 1 octet. Pour utiliser du texte Unicode, il faut passer par l'API « multi-byte », avec les fonctions mbstring (on retrouve les mêmes noms que pour l'API « strings » mais avec le préfixe « mb\_ »). Pour le développeur, la prise en charge d'Unicode devrait faciliter la manipulation des bases de données ou des fichiers XML depuis un script PHP. Pour l'utilisateur, on peut enfin espérer la fin du phénomène « Mojibake », ces fameux bugs d'affichage dans les pages web ou les e-mails. Qui n'a jamais vu des chaînes de caractères tels que :

```
Bonjour C_`dric
```

## Quelques mots sur l'Unicode.

Le but d'Unicode est d'attribuer un nom et un identifiant numérique à tous les caractères, quelle que soit la langue. On compte aujourd'hui environ 245 000 points (signes) définis. La norme Unicode répartit les points dans 17 différents plans :

- Le plus utilisé est le « plan multilingue de base » (PMB). Il contient les points 0x0000 à 0xFFFF (il faut donc 2 octets pour le coder). Ce plan contient les caractères les plus utilisées dans la majorité des textes, c'est-à-dire, entre autres, les alphabets latin, cyril-

lique, grec, arabe, hébreu, ainsi qu'une partie des signes utilisés par le chinois, japonais et coréen.

- Le « plan multilingue complémentaire » (PMC). Il contient les points 0x10000 à 0x1FFFF. Ce plan permet de représenter les caractères rares, comme le gothique, les idéogrammes du linéaire B ou des symboles mathématiques.
- Le « Plan idéographique complémentaire » (PIC) (0x20000 – 0x2FFFF). Il contient des idéogrammes complémentaires pour le chinois, japonais et coréen.

Les autres plans sont soit réservés, soit à usage privé.

L'espace d'Unicode peut contenir un peu plus d'un million de points, codés sur 4 octets.

Il faut comprendre qu'un point ne correspond pas directement à un caractère. Par exemple le glyphe « é » peut être décrit soit directement par son numéro de point, soit par celui du « e » suivi du point « accent aigu ». Ces deux représentations sont strictement équivalentes et doivent être traitées de la même façon. A contrario, certains signes comme la ligature « fi » peuvent être représentés par un seul point Unicode.

Une fois que l'on connaît les différents points Unicode de notre texte, il faut les coder afin de pouvoir les échanger ou en faire une représentation physique en mémoire.

Nous allons présenter les 3 principaux encodages utilisés.

#### UTF-8

C'est un codage à taille variable. Tous les caractères de la table ASCII sont présents dans UTF-8 sans changement. Un texte ASCII est donc déjà converti en UTF-8 (donc sur 1 octet).

Ceci est très important pour la compatibilité avec l'existant et l'adoption de l'Unicode. C'est aussi ce qui explique pourquoi cet encodage est le plus connu aujourd'hui.

Les caractères latin, grec, cyrillique, arabe, hébreu sont codés sur 2 octets. Le reste du plan PMB nécessite au plus 3 octets. Tous les autres caractères nécessitent 4 octets. Un autre avantage d'UTF-8 est qu'il est indépendant de l'endianness de la machine. Par contre, le fait d'être un codage à taille variable complexifie les opérations sur les chaînes de caractères (il faut parcourir la chaîne pour connaître le caractère à la position N). UTF-8 est souvent utilisé pour l'échange de données.

#### UTF-16

Tous les caractères du plan PMB sont codés sur 2 octets. Pour les autres caractères il faut utiliser 2 paires d'octets. Pour l'échange d'information, il est nécessaire d'ajouter une marque pour indiquer l'ordonnancement des octets de la machine (BOM, Byte Order Mark). UTF-16 est souvent utilisé pour les traitements internes à une application.

#### UTF-32

Les identifiants Unicode sont directement codés sur 4 octets. Comme pour UTF-16, un BOM est nécessaire. Cette représentation coûteuse en mémoire n'est utilisée que pour des traitements internes où la rapidité de traitement est importante.

### L'approche choisie par PHP.

Pour intégrer l'Unicode, les développeurs de PHP ont décidé de s'appuyer sur une bibliothèque déjà existante : International Components for Unicode (ICU). Il se trouve que ICU utilise UTF-16 en

Alter Way

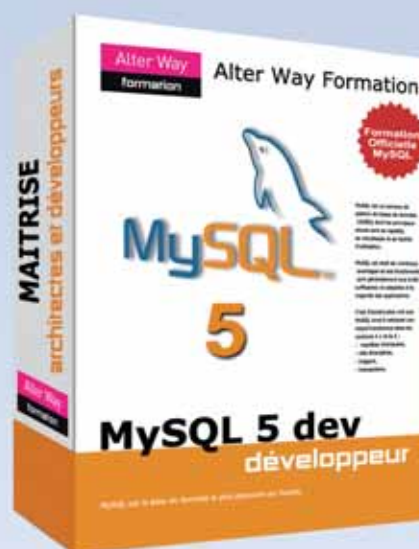
formation

(anciennement Anaska)

## Le Leader de la formation Open Source



### Formation MySQL 5 Développeur



#### Au Programme des 5 jours :

- \*Présentation de MySQL
- \*Requêtes SQL simples
- \*Manipulation des données
- \*Création des bases
- \*Requêtes complexes
- \*Programmation côté serveur
- \*Optimisation

#### Prochaine session :

Paris et Lyon du 28/06 au 02/07

Tarif  
1999€

Tel : 01 41 16 83 70  
[www.alterwayformation.fr](http://www.alterwayformation.fr)  
[formation@alterway.fr](mailto:formation@alterway.fr)



interne, c'est donc naturellement que PHP a aussi choisi UTF-16 pour le codage de toutes les chaînes de caractères. L'idée est de convertir les données une fois en entrée, depuis le système de codage de l'utilisateur vers UTF-16 et de les re-transformer en sortie. Toutes les manipulations internes se font en UTF-16.

Le problème est que les conversions s'avèrent être nombreuses et coûteuses. Dans le monde du web, les données sont souvent en UTF-8 (base de données, fichier XML, ...). Convertir une chaîne depuis UTF-8 vers UTF-16 coûte cher en CPU et en mémoire (dans la plupart des cas, la chaîne UTF-16 sera deux fois plus grande qu'en UTF-8). Ceci complexifie l'implémentation et ajoute des problèmes de compatibilité avec le code existant.

Tout ceci fait que de nombreux développeurs s'éloignent de PHP 6. Certains préféraient adopter une approche plus pragmatique pour résoudre le problème de l'Unicode. Par exemple, une meilleure intégration des fonctions « mbstring », sans changer fondamentalement le moteur. D'autres encore utilisent la branche PHP 5 pour proposer des évolutions du langage, laissant les développements sur PHP 6 stagner.

### Le renouveau

L'année dernière, la branche 5.3 fut créée afin de prendre des éléments du futur PHP 6 et les intégrer dans PHP 5.

En mars 2010, Rasmus Lerdorf décida de déplacer le code qui devait constituer la future version PHP 6 vers une branche et de reprendre le développement principal à partir de la série 5.3. Les évolutions intéressantes pourront toujours être reprises depuis l'ancienne branche PHP 6, qui sera maintenant dédiée à la résolu-

tion du « problème Unicode ». Le but de cette réorganisation est de relancer le développement de PHP qui souffrait d'un manque d'enthousiasme. La communauté PHP est toujours active et va pouvoir recentrer ses efforts sur les évolutions à apporter au langage. Pour ceux qui se demandent à quelle date va sortir PHP 6, cette question n'est pas (n'est plus, N.D.L.R.) à l'ordre du jour. Lorsque suffisamment de fonctionnalités seront développées, une nouvelle version du langage sortira, mais on ne sait pas si ce sera un PHP 5.4, un PHP 6 ou même directement un PHP 7. ... À moins que Perl 6 ne sorte avant :)

### Ressources

Resetting PHP 6

<http://lwn.net/Articles/379909/>

[PHP-DEV] PHP 6

<http://thread.gmane.org/gmane.comp.php.devel/60062>

Minutes PHP Developers Meeting

<http://www.php.net/~derick/meeting-notes.html>

The Absolute Minimum Every Software Developer Absolutely, Positively Must Know About Unicode and Character Sets (No Excuses!)

<http://www.joelonsoftware.com/articles/Unicode.html>

Unicode

<http://fr.wikipedia.org/wiki/Unicode>

■ Cédric Cabessa

Ingénieur développement opensource.

Groupe Devoteam, IDS-Uperto

## L'INFO permanente



- **L'actu** : le fil d'info quotidien de la rédaction
- **La newsletter hebdo** : abonnez-vous, comme 46 000 professionnels déjà. **C'est gratuit !**

## C'est PRATIQUE !

- **Le forum** : modéré par la rédaction et les auteurs de Programmez!, rejoignez les forums techniques de [programmez.com](http://www.programmez.com)
- **Les tutoriels** : une solution en quelques clics !
- **Le téléchargement** : récupérez les nouveautés.

[www.programmez.com](http://www.programmez.com)





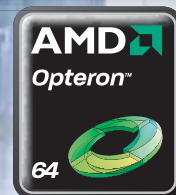
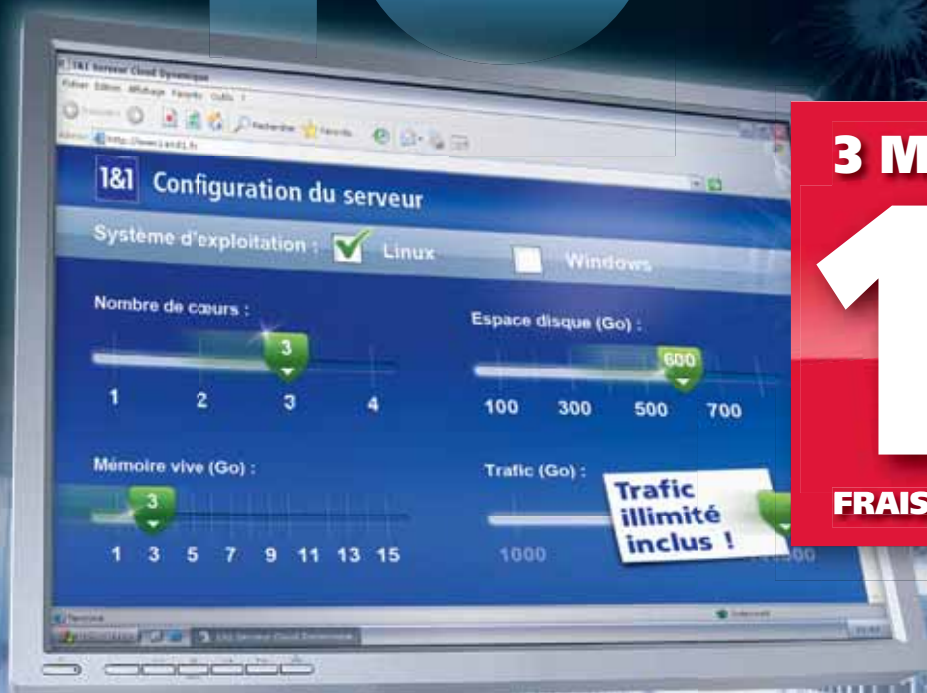
# 10

ANS À VOS CÔTÉS, ÇA SE FÊTE !

3 MOIS À SEULEMENT

# 10€<sup>HT</sup>

FRAIS DE MISE EN SERVICE INCLUS !\*



## 1&1 SERVEUR CLOUD DYNAMIQUE

### Votre serveur sur mesure

Avec le nouveau Serveur Cloud Dynamique, vous ajustez les performances en fonction de vos besoins du moment. Il vous suffit simplement de sélectionner le nombre de cœurs de processeur, l'espace disque et la mémoire vive que vous souhaitez utiliser. Le prix s'adapte automatiquement à la configuration choisie !

Configuration de base (extensible selon vos besoins) :

- ✓ **1 cœur de processeur Quad-Core AMD Opteron™ 2352 (jusqu'à 4 cœurs)**
- ✓ **1 Go de mémoire vive (jusqu'à 15 Go)**
- ✓ **100 Go d'espace disque (jusqu'à 800 Go)**
- ✓ **Trafic illimité**

Consultez toutes nos offres du moment sur notre site Internet !

\*L'offre « 3 mois à 10 € » (10 € HT, soit 11,96 € TTC, frais de mise en service inclus) s'applique à la configuration de base et est soumise à un engagement de 12 mois. Pour une configuration supérieure, vous bénéficiez d'une réduction totale de 82,49 € TTC sur les trois premiers mois. À l'issue des 3 premiers mois, le Serveur Cloud Dynamique est à partir de 23,91 € TTC/mois. Conditions détaillées de l'offre sur [www.1and1.fr](http://www.1and1.fr). Offres sans engagement également disponibles.



Appel non surtaxé

0970 808 911

[www.1and1.fr](http://www.1and1.fr)

# 1&1



# PHP a encore un bel avenir devant lui

La question de la mort de PHP, ou plus simplement de son avenir, est survenue récemment, quand la sortie de PHP 6 s'est vue remise aux calendes grecques.

**M**ais cet arrêt, a priori momentanément, de l'évolution de PHP, n'est dû qu'à des querelles passagères au sein de l'équipe de développement sur la façon de prendre en charge Unicode nativement. Tout cela s'arrangera et PHP continuera encore longtemps. Il suffit de regarder son passé et son présent, pour en déduire un avenir prometteur.

## Le passé

PHP est apparu pour répondre à un besoin simple: générer des pages web dynamiques. PHP n'a pas été le premier à répondre à ce besoin. Ainsi Perl a été longtemps utilisé pour cela dans des scripts CGI. D'autres langages ont été utilisés de manière similaire. Mais à chaque fois, il s'agissait de langages généralistes utilisés

et/ou adaptés pour le web. PHP est, du moins à ma connaissance, le seul langage qui ait été créé pour répondre spécifiquement à ce besoin. Du coup, il est difficile de trouver un langage mieux adapté, ce qui explique son succès dans ce domaine, grandement facilité par le fait que ce langage est Open Source. PHP est à l'inverse des autres, c'est un langage Web qui tend à devenir généraliste :-)

## Le présent

PHP présente de nombreux avantages difficiles à concurrencer. C'est un langage de script, ce qui aide à la rapidité du développement. Il est aussi doté d'une riche bibliothèque parfaitement adaptée. Et PHP est portable. Un langage et des bibliothèques efficaces, c'est bien. Mais dans l'univers du

Web, des frameworks évolués, en l'occurrence des gestionnaires de contenu, c'est mieux. Les CMS les plus populaires sont, sauf erreur de ma part, écrits en PHP. Il en existe de très nombreux, les plus connus étant sans doute Joomla et Drupal. J'utilise ce dernier quotidiennement et il me serait difficile de me passer de sa puissance et de sa souplesse. Enfin, les ténors du web sont construits sur PHP. Je pense par exemple à Wikipedia, et à Facebook. PHP a donc largement prouvé sa viabilité.

## Quelle solution de remplacement ?

Si PHP meurt, dans le monde de l'informatique, cela veut dire qu'il existe une solution de remplacement,

...Suite page 43



## Un langage parfaitement outillé !

Fabien Potencier est une des figures de proue du PHP en France et dans le monde. Il crée dès 2004 le framework Symfony qui est devenu l'une des principales bibliothèques PHP utilisées. Il a bien voulu répondre à nos questions.

**Programmez ! : On fait parfois le reproche à PHP de ne pas posséder de vrais et solides IDE ; est-ce une illusion d'optique ?**

**Fabien Potencier :** Très certainement ! Les développeurs PHP ont un large choix d'IDE. Le plus célèbre est sûrement Eclipse qui propose un support avancé de PHP (le plugin PHP pour Eclipse, PDT, est d'ailleurs le plugin le plus téléchargé du projet Eclipse). NetBeans supporte également PHP de façon exhaustive avec même un support natif pour certains frameworks comme Symfony. Si on ajoute PhpStorm de JetBrains (la société qui édite IntelliJ) qui vient de sortir en version bêta, on voit que tous les grands éditeurs d'IDE ont une offre solide autour PHP. Il existe également les IDE uniquement axés sur PHP comme PHPEdit et Zend Studio. Il me semble donc que PHP

est l'un des seuls langages aujourd'hui à avoir une offre aussi riche en termes d'IDE.

**P ! : Qu'est-ce qui manquerait aujourd'hui à PHP comme fonctions pour le renforcer ?**

**F. P. :** Avec les avancées du langage depuis la sortie de PHP 5.3 l'année dernière, il ne manque pas grand-chose par rapport aux langages concurrents. Le support des "mixins" et/ou de l'héritage multiple serait certainement un plus. La bonne nouvelle, c'est que PHP 5.4 supportera les "traits", un concept qui permet de résoudre ces problèmes de façon très élégante et performante. La maturité du langage en termes de fonctionnalités, ses performances et son approche originale du "share-nothing" ne font que renforcer sa position dominante pour le développement d'applications Web.

**P ! : On parle beaucoup du décalage de PHP 6, des problèmes de planning, qu'en pensez-vous ?**

**F. P. :** PHP 6, tel qu'il avait été défini il y a quelques années, est effectivement abandonné. Mais gardons à l'esprit que toutes les fonctionnalités prévues pour PHP 6, en dehors du support de l'Unicode, ont été intégrées à PHP 5.3.

Nous sommes maintenant dans la phase de spécification et de planification de la prochaine version.

Les discussions étant publiques, tout le monde peut effectivement constater les actuelles tensions de la core team.

Mais c'est tout à fait normal compte tenu de la taille de la communauté.

Tout va rentrer dans l'ordre dans les prochaines semaines.

# 10

**ANS À VOS CÔTÉS, ÇA SE FÊTE !**



**3 MOIS À SEULEMENT**

# 10€<sup>HT</sup>

**FRAIS DE MISE EN SERVICE INCLUS !\***

## **TECHNOLOGIE HEXA-CŒUR**

### **La dernière génération de serveurs 1&1 !**

1&1 vous propose des solutions complètes, sans mauvaise surprise en terme de coûts ! Toutes les fonctionnalités essentielles sont comprises dans le prix de base des nouveaux Serveurs Hexa-Core 1&1.

**Les premiers serveurs dédiés dotés de 12 cœurs de processeur !**

- ✓ **2 processeurs Six-Core AMD Opteron™ 2423 HE**
- ✓ **Jusqu'à 32 Go de mémoire vive**
- ✓ **Jusqu'à 3 To d'espace disque avec RAID 5**
- ✓ **Haute efficacité énergétique**
- ✓ **Meilleur rapport performance/prix**

**Consultez toutes nos offres du moment sur notre site Internet !**

\*L'offre « 3 mois à 10 € » (10 € HT, soit 11,96 € TTC, frais de mise en service inclus) est soumise à un engagement de 12 mois. A l'issue des 3 premiers mois, les Serveurs Dédiés Hexa-Core 1&1 sont à partir de 299,99 € HT/mois (358,79 € TTC). Conditions détaillées de l'offre sur [www.1and1.fr](http://www.1and1.fr). Offres sans engagement également disponibles.



Appel non surtaxé

**0970 808 911**

**[www.1and1.fr](http://www.1and1.fr)**

# 1&1



# Les outils des Décideurs Informatiques

*Vous avez besoin d'info  
sur des sujets d'administration,  
de sécurité, de progiciel,  
de projets ?  
Accédez directement  
à l'information ciblée.*

**L'INFORMATION  
SUR MESURE**



Actu triée par secteur

Cas clients

Avis d'Experts



Actus

Evénements

Newsletter

Etudes  
&  
Statistiques

Infos des SSII

Vidéos

**L'INFORMATION EN CONTINU**



[www.solutions-logiciels.com](http://www.solutions-logiciels.com)



éventuellement meilleure. Mais alors laquelle ? Pendant un temps Python, avec Zope, s'est fait remarquer, mais Zope 3 s'est avéré mal né, et du coup, Python est un peu perdu de vue pour le Web en ce moment, même si des frameworks comme Django mériteraient plus de succès. Ruby on Rails s'est avéré un candidat sérieux. La rapidité de développement procurée par Rails est grandement appréciée, mais Rails n'est qu'un framework dont l'équivalent en PHP pourrait exister a priori. Au niveau du langage, les performances de Ruby semblent en deçà de celles de PHP. Et finalement si Rails a beaucoup fait parler pendant un temps, on a l'impression que l'engouement commence à diminuer, tandis que PHP est toujours là. Nous avons encore les poids lourds: Java et .Net. Java n'a pas particulièrement bonne réputation en ce qui concerne la consommation de ressources, ni en ce qui concerne la facilité de mise en œuvre. Sur ce plan, la plate-forme .Net est peut-être meilleure, et grâce au projet Mono, on la voit apparaître sous Linux. Mais que ce soit pour Java ou Mono, le choix de CMS n'est pas aussi riche que pour PHP. Enfin, les habitudes étant ce qu'elles sont, je ne pense pas que des solutions, pourtant originales et performantes, comme celles basées sur le langage Lisp deviennent populaires un jour. De tout ce qui vient d'être dit, la solution idéale pour remplacer PHP ne se dégage pas vraiment.

## PHP s'impose et est imposé

Le web est un univers où il faut aller vite. Un nouveau site doit pouvoir être mis en place très rapidement, à faible coût, et c'est pour cela que les CMS sont aussi importants. Mais il y

a aussi la question de l'hébergement. De nos jours, tous les hébergeurs proposent du LAMP (Linux Apache MySQL PHP) prêt à l'emploi et à des tarifs intéressants. Ce n'est pas le cas des autres solutions. Bien sûr on peut déployer Ruby on Rails ou une autre solution, sur un serveur dédié, mais ce n'est plus aussi immédiat. On est alors responsable des mises à jour, notamment concernant la sécurité, on doit se tenir au courant et réagir, alors que pour LAMP, un (bon) hébergeur s'en charge, au moins pour le gros du travail. Dans le cas de serveurs mutualisés, on ne discute même plus: l'utilisateur n'a pas les droits de l'administrateur, il doit se contenter des logiciels pré-installés, à savoir LAMP :-)

## L'avenir

PHP s'impose sur le web par ses qualités, et il est imposé par les hébergeurs. Bref il est omniprésent, et comme la solution de remplacement ne saute pas aux yeux, on ne l'imagine pas décliner avant de nombreuses années. En revanche, les choses peuvent évoluer. Comme tout un chacun, PHP présente les inconvénients qui vont avec ses qualités. Ainsi en tant que langage de script, il impose un surcoût à l'exécution que Facebook et ses 30 000 serveurs jugent excessif. Alors Facebook a créé HipHop, un traducteur PHP vers C++. On peut aussi imaginer que des compilateurs PHP verront le jour, pour être utilisés après les phases de mises au point. Ceci semble une évolution probable et intéressante. Mais PHP restera incrusté dans le paysage du web encore longtemps.

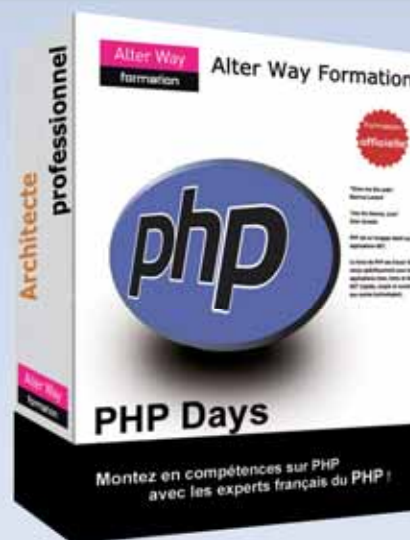


■ Frédéric Mazué

## Le Leader de la formation Open Source



Montez en compétences sur PHP  
avec les experts français du PHP !



### Au Programme :

- \*L'industrialisation
- \*L'environnement de travail
- \*Les bonnes pratiques de la programmation
- \*Gestion de projet et méthodes

### Détails pratiques :

Paris du 08/07 au 09/07

Tarif  
1090€

Plus d'infos sur : [www.phpdays.com](http://www.phpdays.com)

Tel : 01 41 16 83 70  
[www.alterwayformation.fr](http://www.alterwayformation.fr)  
[formation@alterway.fr](mailto:formation@alterway.fr)



# Windows Azure : le cloud computing pour les développeurs



## Windows Azure

Depuis début 2010, l'environnement cloud computing de Microsoft, Windows Azure est disponible à la commercialisation. Depuis avril, c'est l'outil AppFabric qui a été finalisé et avec Visual Studio 2010, le développeur bénéficie d'heures Azure incluses, plusieurs centaines d'heures ! De quoi, enfin s'amuser dans le nuage...

Azure est une plateforme cloud complète tournant sur une infrastructure Windows

Server / Hyper-V que l'on ne voit pas. Tout est automatique, le plus transparent possible pour l'utilisateur. L'objectif ? Exécuter ses applications .Net, Java, PHP, Python, directement sur Internet, ou plutôt le nuage. Il s'agit de faire le minimum de modifications à son application pour qu'elle puisse en quelques heures fonctionner sur le nuage. Azure mise sur l'interopérabilité vers les autres langages pour attirer les développeurs.



(SQL Azure), ce qui n'est pas toujours le cas dans les offres concurrentes. Le modèle de développement se veut le plus proche possible de .Net, même s'il faut tout de même considérer les contraintes et les spécificités du cloud. Mais le 100 % cloud n'est pas pour demain. L'approche S+S de Microsoft, moquée par la concurrence et désormais acceptée, permet une transition en douceur entre le modèle classique client / serveur et le cloud, et surtout, on peut ainsi mixer les composants restés chez soi et les composants sur le cloud.

Nous ne sommes qu'au début de l'évolution. Le cloud demeure un marché très jeune avec une multitude de fournisseurs et très peu d'interopérabilité entre eux. Mais l'offre se structure rapidement, même si les spécifications communes à tout le monde tardent à être discutées, elles le seront d'une manière ou d'une autre. Assurément, le cloud est un des avènements du marché logiciel et donc du développement. Il ne faut pas céder à la fausse vérité de certains analystes qui voient d'ici 2015 un cloud computing partout. Non, le cloud restera marginal dans les 5 prochaines années (15-20 % du marché logiciel global) mais il ne fera que croître. C'est dès aujourd'hui qu'il faut regarder, comprendre le cloud. Soyez un développeur Azure ready ! Notre dossier du mois va vous permettre de voir le potentiel d'Azure, son modèle de développement, les aspects données, la migration, les outils ou encore le modèle économique.

Bon voyage au cœur du cloud !

■ François Tonic

Récemment, Salesforce a annoncé pour 2010 une plateforme Java pour Force.com avec l'aide de VMware et le framework Spring. L'un des enjeux du cloud est de pouvoir exécuter le code actuel... Mais le pari est loin d'être gagné pour de nombreux langages.

Azure arrive avec des SDK, des API, des services de données, de sécurité, d'administration. On dispose par exemple d'un vrai moteur de base de données relationnelle

# L'architecture de Windows Azure

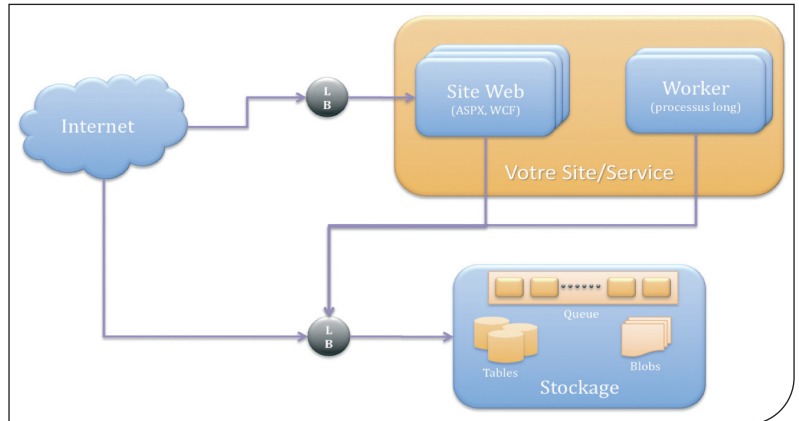
Windows Azure est la plateforme cloud computing, c'est-à-dire disponible via Internet. On cible bien ici la plateforme, dans le monde cloud on parlera de PaaS : Platform as a service. C'est la couche d'exécution et de développement du cloud, au-dessus de l'infrastructure cloud (IaaS) mais en-dessous de la couche purement applicative (SaaS).

**C**omme Azure est une plateforme, on dispose de services, d'ensemble de services : AppFabric, SQL Azure, Azure SDK, Azure Drive, etc. Cela signifie aussi qu'il est nécessaire de modifier, d'adapter l'architecture logicielle de son code pour pouvoir le faire fonctionner en mode Azure. Fondamentalement, nous disposons de deux rôles : le Web Role et le Worker Role. Pour rappel, SQL Azure est une base de données relationnelle, AppFabric est une couche de connectivité et de fédération d'identité. À noter qu'Azure propose son propre mécanisme de stockage si le relationnel n'est pas indispensable dans votre projet.

## Le rôle des rôles

Si nous reprenons les propos de MSDN, voici comment les deux rôles sont présentés :

- Un rôle Web est une application Web qui s'exécute sur les services Internet (IIS). Il est accessible par le biais d'un point de terminaison HTTP ou HTTPS.
- Un rôle worker est une application de traitement d'arrière-plan qui exé-



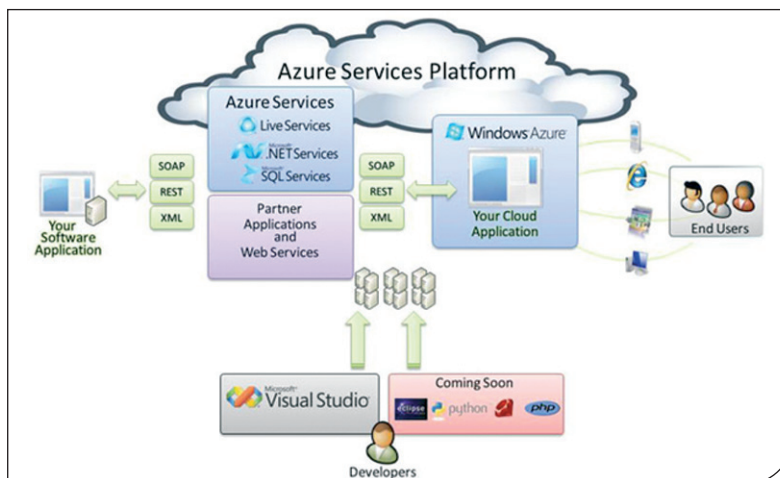
cute du code .NET arbitraire. Il a également la capacité à exposer des points de terminaison internes et ouverts sur Internet.

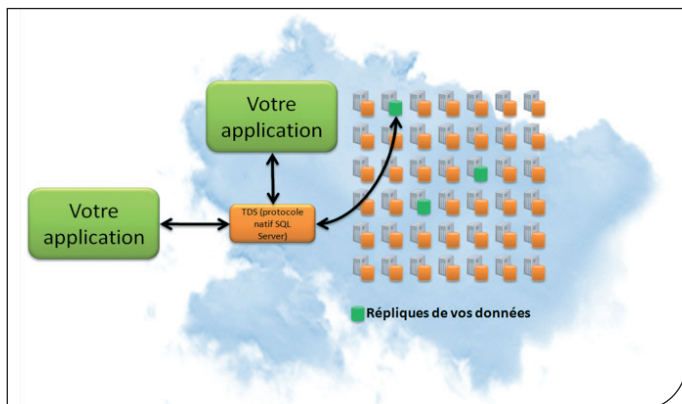
- Un VM role apparaîtra très bientôt...

Un service Azure doit inclure un de ces deux rôles mais en réalité, une application Azure est constituée d'un certain nombre de ces rôles. Par exemple, un web role peut implémenter dans mon service cloud un site web auquel les utilisateurs (dudit service) peuvent accéder via une URL de type : <http://monservice.cloudapp.net>. Et en arrière-fond, un worker role va traiter les informations utilisées par

mon web role. Par définition, ces rôles web sont des projets d'application ASP.Net mais ils comportent des caractéristiques cloud avec la présence des API propres à Azure pour le diagnostic et la journalisation, API d'environnement et de runtime ou encore pour le stockage des données. Notez que le web role peut aussi fonctionner avec du PHP, Python, Ruby, WCF... Le tout se fait en sécurisation SSL. Sur le worker role, notons qu'il peut être avec ou sans connexion entrante. Il peut se comparer à un daemon Windows (= Windows Service). L'accès à l'environnement Azure se réalise par le RoleEnvironnement. Il permet d'accéder à la configuration du service cloud, aux ressources locales (disque local), aux informations du rôle et expose aussi des événements.

SQL Azure est une base de données relationnelle hébergée dans les datacenters Microsoft. Sa maintenance et sa redondance sont assurées directement par Microsoft. Comme elle utilise les protocoles normalisés de SQL Server, vous pouvez vous appuyer dessus indifféremment depuis une application située dans l'entreprise ou dans le Cloud. Large-





ment compatible avec SQL Server 2008, SQL Azure permet de répliquer et d'enrichir les bases de données d'entreprise de façon simple. Attention tout de même, aujourd'hui, SQL Azure est limité par rapport à SQL Server, notamment sur la sécurité ou certaines fonctions accédant aux couches systèmes. Azure Storage est l'autre composant stockage d'Azure. Il se décompose en trois parties : blob, table et queue. Pour simplifier l'intégration entre les serveurs d'entreprise et le Cloud, les services d'AppFabric apportent les capacités fondamentales pour l'intégration d'applications :

## • Connectivité

L'Internet Service Bus est la composante de AppFabric permettant la mise en relation de serveurs et d'applications indépendamment de leur lieu d'hébergement, offrant notamment la connectivité transparente entre applications situées derrière des firewalls, de façon sécurisée et interopérable

## • Sécurité et fédération d'identité

Afin de permettre l'accès aux applications et services entre différentes entreprises, AppFa-

abric Access Control Service fournit une passerelle de sécurité qui sert d'intermédiaire entre les applications et les utilisateurs. Cette passerelle est compatible avec Active Directory ainsi qu'avec les standards de gestion d'identité sur Internet (WS-Trust, WS-Federation, SAML, mais aussi Live Id et OpenId)

## Un système distribué

Pour bien comprendre Azure, il faut avoir à l'esprit que le système n'est pas disponible en boîte vendue dans le commerce et encore moins installable chez soi. Non, Azure est un système massivement distribué dans les datacenters de Microsoft. C'est-à-dire qu'il fonctionne sur des centaines de milliers de serveurs à travers le monde. Azure couvre la partie infrastructure qui demeure ainsi invisible. L'infrastructure s'appuie sur du Windows Server 2008, HyperV, IIS7, les langages de développements, .Net, etc.

Azure comme d'autres solutions du même type permet d'avoir une vraie informatique à la demande, capable de montées en charge très fortes et de s'adapter à vos besoins.

■ François Tonic

## Liens indispensables

<http://msdn.microsoft.com/fr-fr/azure/msdn.coach.azure.aspx>  
<http://blogs.developpeur.org/redo/archive/2010/01/31/presentation-et-demos-du-windows-azure-roadshow.aspx>  
<http://windows.azure.com>

# Aastra

Concours Européen  
de développement  
d'applications XML  
sur terminaux  
SIP Aastra

## Concours XML 4 SIP

Développez des applications XML  
sur terminaux SIP Aastra

Du 15 avril au 30 octobre 2010

Ouvert à tous les développeurs

Jury composé de 13 experts

Quatre catégories d'applications

- ★ Web 2.0,
- ★ Entreprise,
- ★ Téléphonie,
- ★ Vie pratique.



Aastra 6739i

De nombreux lots à gagner  
pour récompenser les meilleures applications

## Inscrivez vous

Pour en savoir plus sur le concours XML 4 SIP  
et vous inscrire :

[www.aastra.fr/XMLcontest/](http://www.aastra.fr/XMLcontest/)



# Le prix de l'Azure

Une des problématiques fondamentales du cloud est de connaître son coût. Car le cloud, qu'il s'agisse d'Azure, Amazon, Google, a un prix. Mais la tarification cloud n'est pas facile à comprendre et à maîtriser. Essayons de nous y retrouver.

**A** la base, il existe 3 offres tarifaires Azure : accélérateur de développement de base de la plateforme Windows Azure, accélérateur de développement étendu de la plateforme Windows Azure et consommation sur la plateforme Windows Azure. À cela se rajoute une offre spéciale découverte, gratuite. Mais avant de décrire ces offres, voyons tout d'abord de quoi se compose la tarification Azure.

## Un ensemble de tarifs à inclure

Pour calculer le coût d'Azure, il faut considérer plusieurs éléments : le

temps de compute (les instances), le stockage, SQL Azure, AppFabric, les transferts de données. Cela se traduit de la manière suivante :

## Le temps de compute

Il s'agit du temps d'utilisation d'une instance Azure par heure d'utilisation. Le coût varie de 0,0852 € à 0,6809 €. Cette variation est due à la différence entre les types d'instance. La petite instance comprend un serveur à 1,6 Ghz, 1,75 Go de mémoire, 225 Go de stockage d'instance et des performances E/S dites modérées. L'extra large comprend 8 \* 1,6 Ghz, 14 Go de mémoire, 2 To de stockage et des performances élevées. Il faut donc adopter l'instance la plus adaptée à ses besoins. Attention : même si vous n'utilisez que 10 minutes d'une heure, une heure entière sera facturée.

## Stockage

Coût du stockage des données par mois et par Go consommé. Il faut aussi inclure le coût des transactions (par paquet de 10 000 transactions de stockage). Soit :

- 0,1064 € mois / Go
- 0,0071 € / 10 000 transactions de stockage

## SQL Azure

L'utilisation de SQL Azure sera elle aussi facturée selon deux modèles : web édition (maximum 1 Go de données relationnelle) et business édition (10 Go). Le coût est le suivant :

- 7,085 € par base et par mois en web édition
- 70,913 € par base et par mois en business édition

Windows Azure Storage est facturé en fonction de l'utilisation moyenne du stockage par blob, tableau, file d'attente et disque au cours d'une période de facturation. Par exemple,

une utilisation de 10 Go de stockage au cours de la première moitié du mois et rien lors de la seconde, seront facturées à hauteur de la moyenne de votre utilisation du stockage, soit 5 Go.

## AppFabric : l'usine à applications Azure

L'environnement AppFabric permet de faciliter la conception d'applications Azure. Deux éléments sont facturés : l'access control à hauteur de 1,4114 € pour 100 000 transactions et le service bus (2,8298 € par connexion, pack de connexions disponibles).

## Transfert de données entre les datacenters

Comme pour d'autres fournisseurs de cloud, Azure facture les transferts de données entrants et sortants. Le prix varie selon la région du monde. Pour l'Europe :

- 0,071 € Go / entrant
- 0,1064 € Go / sortant

A noter qu'Azure Drive n'est pas facturé pour le moment. Pour faire du PRA, il faut disposer de deux instances (minimum).

■ François Tonic

## LES OFFRES AZURE POUR DÉVELOPPEUR ET ENTREPRISE

### Découverte de la plateforme Windows Azure

Cette offre gratuite comprend 25 heures de compute (petite instance), 500 Mo de stockage, 10 000 transactions de stockage, 1 base SQL Azure (gratuit pendant 3 mois), AppFabric avec 100 000 transactions access control et 2 connexions Service bus. Elle inclut 500 Mo de transferts en données sortantes et entrantes. En coût mensuel (sur les 3 premiers mois, car au-delà, il faut inclure la base SQL Azure), l'offre découverte équivaut à environ 18,4231 €.

### Pack accélérateur de développements standard

Cette offre coûte 42,52 € par mois d'abonnement (base 6 mois). Des coûts de dépassements sont prévus. L'offre comprend 750 heures de compute, 10 Go de stockage, 1 000 000 de transactions (idem pour access control), 5 connexions service bus, entre 7 et 14 Go de transferts de données. Pour une utilisation assez soutenue cette offre est un bon compromis.

### Pack accélérateur de développements étendu

Offre à 77,98 € par mois (abonnement initial de 6 mois). Elle comprend les mêmes caractéristiques que l'offre standard mais inclut par défaut jusqu'à 10 Go de données pour la base de données relationnelle.

### Pack consommation de la plateforme

On paie à la demande la consommation effective d'Azure. Selon un tarif défini par Microsoft.

## Du temps Azure pour les abonnés MSDN

Pour favoriser Azure auprès des développeurs, les abonnés MSDN peuvent utiliser le cloud de Microsoft sans frais supplémentaires. Une offre de lancement valable 8 mois comprend 750 heures par mois de compute en petite instance soit l'équivalent de 1264 \$ (niveau premium, ultimate et BizSpark).

# Créer son premier service Windows Azure

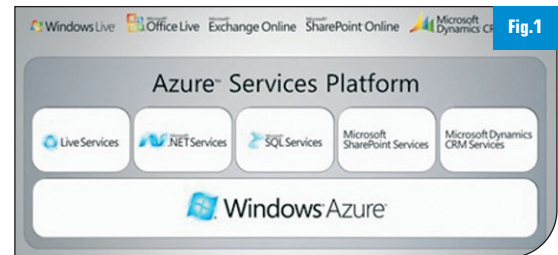
Cet article a pour but de vous montrer comment créer un service pour Windows Azure de A à Z. Certains prérequis sont nécessaires comme la connaissance de Visual Studio 2008 ou 2010. Mais aussi, il faut savoir faire la différence entre une application de type Azure et une application de type Mesh. Il paraît évident que l'application Cloud que nous souhaitons créer est de type Azure compte tenu du sujet. Néanmoins, il n'est pas inutile de garder à l'esprit la différence entre ces deux types.

**A** ce stade, il y a une différence notable car Live Mesh est comme nous le montre le schéma fig. 1, un élément constituant de la plateforme Windows Azure Services qui repose donc sur Windows Azure. De plus, Live Mesh n'est ni plus ni moins qu'une sorte de plateforme de partage de dossiers ou documents mais permettant aussi l'hébergement d'applications de type MeshApps. Du côté de Windows Azure on parle plus de système d'exploitation, sur internet. Ce service a pour but de répondre à des besoins importants de calculs et de traitements, d'accès géo localisé, de stockage, ou même encore de gestion des pics de connexion avec la montée en charge.

Concernant les outils dont nous avons besoin ici, nous allons procéder en deux temps. D'abord, nous allons développer notre service de manière locale. Pour cela, il vous faut Visual Studio 2008 d'installé avec le **Windows Azure SDK** ainsi que les **Windows Azure tools for Microsoft Visual Studio**. Ensuite, une fois que notre service sera fonctionnel en local, sur notre machine, nous allons pouvoir le déployer sur le cloud de Windows Azure. Pour pouvoir effectuer ce déploiement, il vous faut un

compte Windows Live ID et vous rendre sur la page suivante : <http://windows.azure.com/> pour lier votre compte Windows Live ID au portail Windows Azure. Une fois votre compte lié, il vous faudra créer votre compte d'accès à la plateforme Windows Azure en entrant une clé d'activation si vous en possédez une, ou en vous rendant sur le site de Microsoft afin de prendre un abonnement pour Windows Azure.

Nous pouvons maintenant commencer. D'abord, lancez Visual Studio en mode administrateur et créez un projet de type « **Windows Azure Cloud Services** » que l'on nomme ici « **MonPremierWebRole** ». Une boîte de dialogue apparaît alors, on va pouvoir sélectionner les rôles que l'on souhaite ajouter à notre service. Il existe deux types de rôles : les **Web roles** et les **Worker roles**. Un Web Role est en fait une application Web accessible par protocole HTTP et/ou HTTPS, il permet donc le support d'application de type ASP.NET ou Windows Communication Foundation. Un Worker Role est quant à lui une application de type processus, il permet de gérer la communication avec les services de stockage (comme les SQL Services par exemple) ou tout autre service internet. Nous sélectionnons

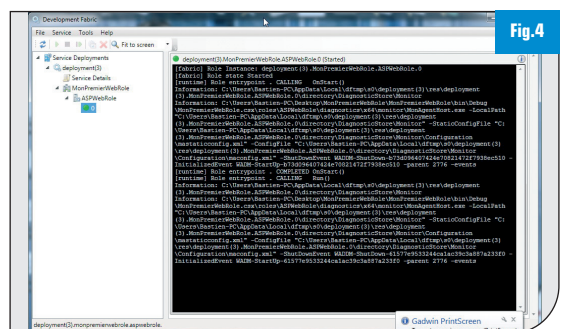
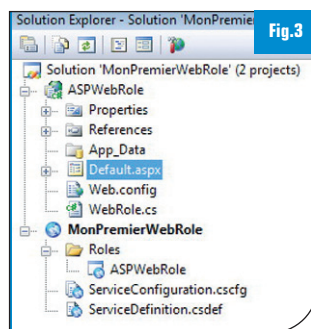
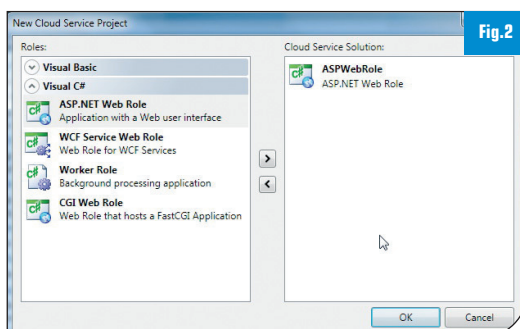


ici un ASP.NET Web rôle que l'on nomme « **ASPWebRole** » [Fig.2].

## Création du service Windows Azure

On se retrouve alors dans Visual Studio après la création du projet avec un explorateur de solution comprenant deux types de projets.

Notre Web rôle ASP.NET de type application ASP.NET, et notre projet Windows Azure Cloud Services contenant les références des rôles que l'on souhaite pour notre service Azure ainsi que deux types de fichier : le **ServiceConfiguration** (qui contient les paramètres de configuration pour nos rôles ainsi que le nombre d'instances que l'on souhaite pour nos rôles en cas de pics de charge) et le **ServiceDefinition** (qui contient les endpoints sur lesquels on veut exposer notre website). Il suffit donc d'utiliser le projet ASP.NET comme un projet classique et de mettre en place tout son



site web. Pour notre exemple nous mettrons seulement une image de fond avec un texte [Fig.3].

Compilez l'application en mode local. Une fenêtre exécutant la Development Fabric (l'équivalent de la Fabric Azure) va s'ouvrir. La **Development fabric** simule l'environnement Azure sur notre poste local. On peut voir sur celle-ci notre déploiement contenant l'ASP.NET web rôle qui contient, lui, une instance. On peut alors voir les « traces » de l'exécution de cette instance. On constate également qu'un gestionnaire internet s'est lancé en mode localhost avec notre web rôle asp.net. Notre application est fonctionnelle de manière locale, nous allons maintenant la déployer sur le cloud de Windows Azure [Fig.4].

## Déploiement du service sur la plateforme Windows Azure

Avant de pouvoir hoster son application sur la plateforme Azure, nous allons faire un publish mais attention nous devons l'effectuer sur le projet Windows Azure Cloud Services. Clic droit sur le projet → publish. Cela va ouvrir le gestionnaire internet sur le site d'administration de la plateforme Windows Azure ainsi qu'un explorer avec notre package « *MonPremierWebRole.cspkg* » compilé et un fichier « *ServiceConfiguration.cscfg* ». Le fichier « *cspkg* » est une sorte de fichier compressé contenant les éléments de notre site web qui seront à déployer c'est-à-dire tous les fichiers de configuration et de définition, ainsi que notre rôle. Le fichier « *cscfg* » contient lui la configuration de notre service. Cette configuration pourra

être modifiée une fois notre service hébergé sur la plateforme Azure [Fig.5]. On peut maintenant créer un nouveau service. On clique pour cela sur notre projet puis sur New Service. Nous avons le choix entre « **Storage account** » (pour le blob, la table et les queues) et « **Hosted service** » (pour hoster nos Web ou Worker Roles). On crée donc un Hosted service [Fig.6].

Il s'agit ensuite de nommer notre service, faire une description, sélectionner une région précise pour l'exécution de notre service et enfin de créer le service.

On va maintenant pouvoir déployer notre application. Pour cela, nous avons le choix entre deux environnements : un environnement de **staging** (met à disposition notre application sur un domaine généré à la volée afin que nous puissions tester son fonctionnement) et un environnement de **production** [Fig.7].

Nous déploierons donc dans un premier temps notre application en environnement de staging. Pour cela nous allons uploader notre ApplicationPackage et notre ServiceConfiguration et nommer notre déploiement. Première étape, Windows Azure va préparer notre machine virtuelle qui contient notre application ASP.NET et va la déployer sur différents serveurs physiques prêts à être virtualisés.

Quelques minutes plus tard, on peut lancer la virtualisation de notre application en cliquant simplement sur le bouton « **Run** ». Windows Azure démarre alors une nouvelle machine virtuelle qui nous est entièrement dédiée, sur laquelle va être déployée notre application : cette opération peut donc prendre un certain temps, entre 5 et 15 minutes. L'application est en mode **d'initialisation** [Fig.8].

Encore quelques minutes plus tard, le Web Rôle se retrouve en mode « **Started** » et est accessible sur une adresse temporaire de type

<Guid>.cloudapp.net (puisque nous sommes dans l'environnement de staging). Remarquez la présence du bouton « **Configure** », il permet de modifier les informations du fichier ServiceConfiguration : les informations de notre service.

Une fois notre application testée et fonctionnelle en environnement de staging, nous allons pouvoir la faire passer en environnement de production via le bouton circulaire du milieu. Si vous désirez remplacer cette application déployée par une autre, vous devrez d'abord suspendre l'application démarrée (via le bouton « **Suspend** ») afin de pouvoir ensuite la supprimer (via le bouton « **Delete** ») et déployer une nouvelle application sur le cloud de Windows Azure [Fig.9].

## Conclusion

Comme vous avez pu le constater, créer un service Azure n'est pas compliqué. D'abord parce que nous utilisons nos outils de développement familiers (Visual Studio). Aussi, parce que nous avons à notre disposition un environnement d'exécution et de debugging local grâce à la Development Fabric.

■ Bastien Chauvin  
■ Thibault Laurens  
*Dotnet france*

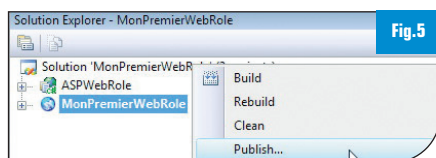


Fig.5

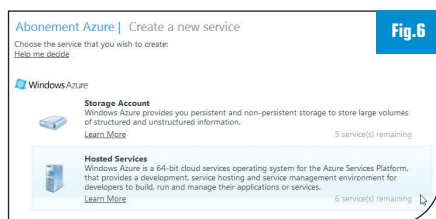


Fig.6

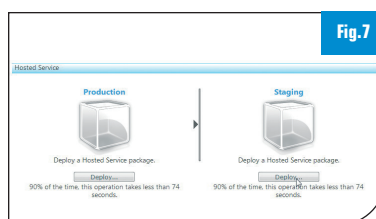


Fig.7

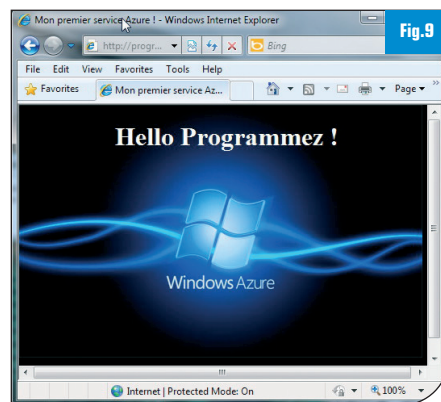


Fig.9

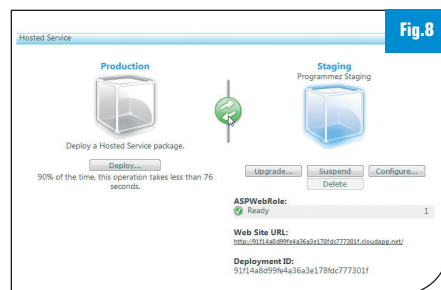


Fig.8



# Le stockage dans le nuage Azure

Depuis le 9 janvier 2010, Microsoft a rendu disponible sa nouvelle plateforme dite « On the Cloud » nommée Azure. Elle présente de nombreux avantages et services (la dématérialisation totale des ressources, la montée en charge...) sans avoir à gérer l'infrastructure sous-jacente.

**C**ela permet de choisir à la volée la localisation du Datacenter qui stockera potentiellement les données (à savoir qu'il existe actuellement 6 Datacenters dans le monde). Ce système est dit « flexible », l'accès peut être facilité via l'adressage par Url (les « endpoints ») de tout élément :



Nous montrerons dans les parties suivantes comment exploiter la souplesse de ce service Microsoft.

## Les pré-requis

Si vous décidez de développer une solution avec l'environnement de Visual studio (2008 ou 2010), il vous faudra tout d'abord le Windows Azure SDK téléchargeable à cette adresse : <http://www.microsoft.com/windowsazure/getstarted/>

L'installation nécessite en premier lieu un IIS 7, installé sur votre OS, ainsi que l'activation de features Windows de développement tels que l'ASP.NET (rajout automatique des extensions .NET ainsi que des deux ISAPI).

**Remarque :** certains outils vous faciliteront le développement en vous affichant l'arborescence de votre Storage, comme Azure Storage Explorer, téléchargeable sur CodePlex à l'adresse : <http://azurestorageexplorer.codeplex.com/>

Les services du SDK pour SQL Azure s'installent en utilisant l'instance par défaut SQLEXPRESS. Cependant, si vous utilisez une autre instance à la compilation, vous aurez une erreur. Pour pallier ce problème, l'utilisation du Prompt Azure SDK sera nécessaire avec la commande :

```
Windows Azure SDK Shell
C:\Program Files\Windows Azure SDK\v1.1>DSinit /sqlinstance:votreinstance
```

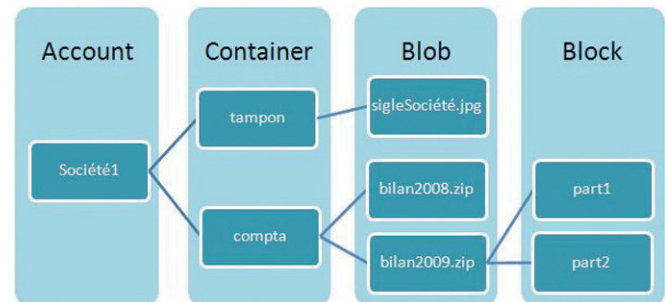
## Windows Azure : type Storage et type Compute

La partie Compute de Windows Azure délivre la puissance de calcul CPU, donc la mise en place et l'exécution des services clients que l'on nommera Cloudservices (WebRole et WorkerRole). La deuxième partie, le Storage, est assimilable à une solution de stockage. Celle-ci se divise en trois catégories : le Blob service, le Table service et le Queue service. L'accès au Storage se fait par protocole REST.

### Les Azure Blobs :

Les Azure Blobs (présentés comme des unités) sont rassemblés dans un container assimilable à un dossier. Ces Blobs permettent le stockage de différents types de fichiers, qu'ils soient de type

multimédia, texte, archive, etc. Account englobe tous les containers d'un compte spécifique.



On remarquera qu'il existe deux spécificités, les Blobs « block » ou « page ». Les premiers sont optimisés pour le streaming, par exemple : une vidéo pesant 400 Mo sera stockée en 100 blocks de 4 Mo chacun. L'utilisation des « pages », quant à elle, sert pour la lecture/écriture aléatoire, ex : écriture d'un agenda. Un Blob peut être accédé par son url de type : <http://<account>.blob.core.windows.net/<container>/<blobname>>

Pour accéder à un Storage sur le « Cloud », le SDK nous fournit des classes telles que CloudBlobClient, CloudBlobContainer ou encore BlobContainerPermission. Ces accès ne peuvent être effectués qu'à l'aide d'une connexion à votre compte Azure, par le biais d'une instance de classe de type CloudStorageAccount.

```
// création de la connexion au Storage par le biais du fichier ServiceDefinition.csdef et ServiceConfiguration.cscfg
var storageAccount = CloudStorageAccount.FromConfigurationSetting("StorageAccountConnectionString");
CloudBlobClient blobStorage = storageAccount.CreateCloudBlobClient();

//Récupération de l'object Container à partir du Storage
CloudBlobContainer container = blobStorage.GetContainerReference("compta");
container.CreateIfNotExist();

//Mise en place des permissions (dans ce cas, publique)
BlobContainerPermissions permission = container.GetPermissions();
permission.PublicAccess = BlobContainerPublicAccessType.Container;
container.SetPermissions(permission);
```

Un exemple d'ajout et de suppression :

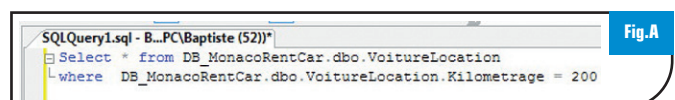
```
//Création d'un nom pour chaque photo
var name = DateTime.Now.Millisecond.ToString();

CloudBlobClient blobStorage;

//Récupération du container visé
CloudBlobContainer container = blobStorage.GetContainerReference("compta");
//Ajout de la photo par un blob
CloudBlockBlob blob = container.GetBlockBlobReference(name);
blob.UploadFromStream(FileUploadPhoto.FileContent);
```

### Les Azure Tables :

Gérées par le « Table service », les Azure tables de type clé/valeur sont considérées comme des bases de données sans la logique relationnelle que l'on accorde vraiment aux bases de données MSSQL (ne possédant pas de clés primaires et de clés étrangères au sens propre). Vu que les ressources sont dématérialisées, une table peut très bien se retrouver sur plusieurs serveurs (voire des centaines) une requête simple [Fig.A] questionnera, serveur après



serveur, notre table. Avec une requête utilisant des « PartitionKey », et des « RowKey », la recherche est plus rapide puisque ce couple de propriétés forme la clé primaire d'identification qui permet la récupération directe sur le serveur concerné. A savoir qu'il existe une propriété Timestamp de type DateTime qui enregistre côté serveur la date du dernier accès à une entité. Cette propriété est utilisée en interne pour l'optimisation, vous ne pourrez donc pas l'utiliser. Une entité peut être accédée par son url de type : <http://<account>.table.core.windows.net/customer?PartitionKey= 'partitionkey', RowKey='rowkey'>).

Pour accéder à une Table, on utilisera des classes telles que CloudTableClient ou TableServiceContext. On utilisera les mêmes identifiants de connexion au Storage.

```
const string Table = "MyTableEntity";
//Création de l'objet utilisant les identifiants de connexion
var storageAccount = CloudStorageAccount.FromConfigurationSetting("StorageAccountConnectionString");
//Création d'un objet de type CloudTableStorage, utilisant le Endpoint de votre compte azure ( ici http://127.0.0.1:1002/ ),
// ainsi que les identifiants de connexion.
CloudTableClient tableStorageClient = new CloudTableClient(storageAccount.TableEndpoint.ToString(), storageAccount.Credentials);
//Création de l'objet Context de type TableServiceContext
var Context = tableStorageClient.GetDataServiceContext();
tableStorageClient.CreateTableIfNotExists(Table);
```

Un exemple de requête sur une Table

```
//Requete permettant de recuperer une liste d'objet de type TableRow ( notre objet à stocker )
var query = (from tbr in Context.CreateQuery<TableRow>(Table)
where tbr.Processed == true
select tbr).AsTableServiceQuery();

//Enumération par PartitionKey
IEnumerable<TableRow> rows = query.Execute().OrderByDescending(m => m.PartitionKey);
List<TableRow> list = new List<TableRow>(rows);
```

## Les Azure Queues :

Les Azure queues jouent le rôle d'une pile FIFO pour permettre le traitement asynchrone des messages (d'une taille maximale de 8 Ko). L'utilisation va souvent de paire avec la prise en charge d'un WorkerRole. Ce dernier peut lire et traiter le message, le supprimer ou le renvoyer dans une Queue. Si l'opération échoue, le message est remis dans la queue d'origine. Le message est accessible dans la queue via l'url de type :

<http://<account>.queue.core.windows.net/<queueName>/messages>.

Pour utiliser une queue, sont mises à disposition des classes telles que CloudQueueClient, CloudQueue, ou encore CloudQueueMessage, de façon à générer un message mis en attente dans sa pile FIFO :

```
// Récupération du Storage
var storageAccount = CloudStorageAccount.FromConfigurationSetting("StorageAccountConnectionString");
// Création d'un client d'accès utilisant les queues
CloudQueueClient queueStorage = storageAccount.CreateCloudQueueClient();
//Création d'une queue
CloudQueue queue = queueStorage.GetQueueReference("objectAutorisé");
queue.CreateIfNotExists();
//Création d'un message destiné à la queue ayant pour contenu la valeur d'une TextBox
CloudQueueMessage queue2 = GetThumbnailMakerQueue();
queue2.AddMessage(new CloudQueueMessage(System.Text.Encoding.UTF8.GetBytes(txtName.Text)));
```

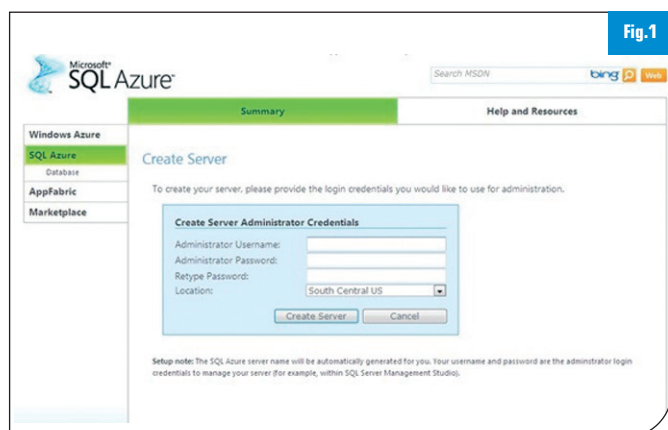


Fig.1

## SQL Azure, une base dématérialisée

SQL Azure comme son nom l'indique est un service de la plateforme Microsoft Azure. On peut le deviner, SQL Azure est un service « on the Cloud » hébergeant des bases de données relationnelles, construit sur la fondation SQL Server. Son utilisation se justifie par exemple dans le cas de la création d'un Webservice qui aurait besoin de relations complexes entre ses données. Lorsque les données sont partagées ou lorsqu'elles sont réparties sur plusieurs sites, la solution SQL Azure peut être judicieuse également.

### SQL Azure, où sont les avantages ?

Avec la conception de bases dématérialisées, SQL Azure vous affranchit d'un maximum de tâches d'installation/administration des serveurs qui sont dorénavant gérées par la plateforme. Le souci de sauvegarde est résolu par la mise en place automatique de backups qui répliquent vos bases sur d'autres serveurs, ce qui assure une haute disponibilité ainsi qu'une garantie contre la perte de vos données. Tout problème est géré en interne, ce qui vous permet une utilisation transparente, en cas de panne sur un serveur, un autre le remplacera immédiatement. Un autre avantage est celui de l'expérience très symétrique pour le développement et l'utilisation des outils (utilise TDS protocol et T-SQL). Enfin, l'atout principal de SQL Azure est qu'il permet la montée en charge de façon très simple et rapide. Selon la nécessité, des ressources supplémentaires sont attribuées ou libérées. Cette adaptation vous assure un coût, le plus juste possible, avec le paiement à l'usage et votre budget ne souffre pas d'une mauvaise estimation de vos besoins.

Côté sécurité, SQL Azure utilise l'authentification SQL (UID/PWD). Le modèle d'autorisation est entièrement compatible avec SQL Server. Les rôles Admin ont les permissions : CREATE/DROP database, CREATE/DROP/ALTER login, GRANT/REVOKE rights, modifier les settings firewall.

### Une base de données en quelques clics

Pour mettre en place la solution SQL Azure proposée par Microsoft, vous devez tout d'abord « créer votre serveur virtuel ». Remplissez le formulaire en définissant un nom d'administrateur (à savoir que certains noms génériques ont été bannis, comme Admin etc.), un mot de passe et une localisation, qui sera le Data-center de votre serveur [Fig.1].

Vous venez de finir la création de votre serveur, vous accédez

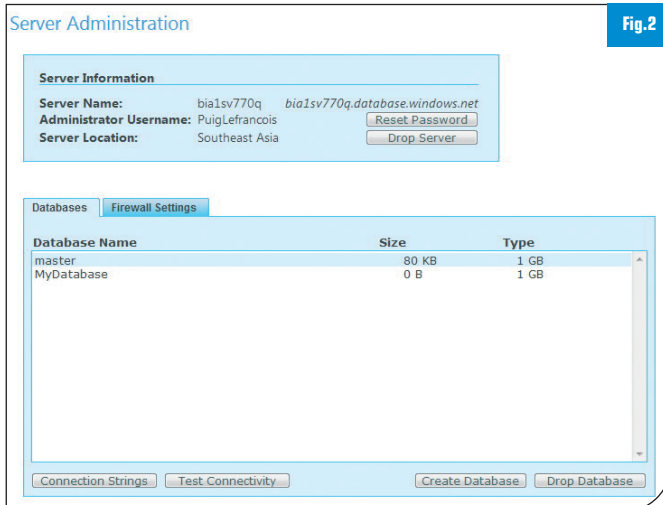
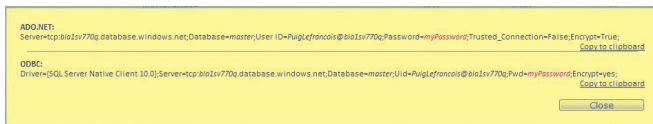


Fig.2

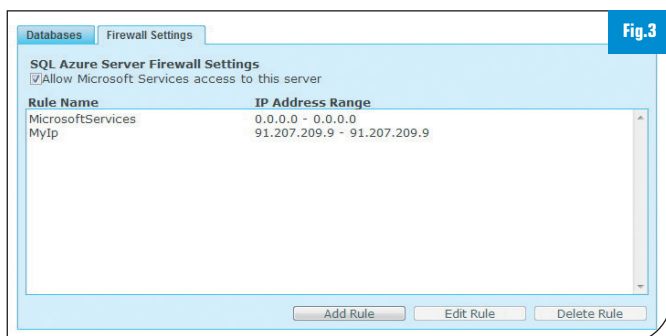
maintenant à l'interface d'administration de votre service SQL Azure. Dans le premier encadré vous retrouvez les informations relatives à votre compte. On peut dès lors référencer avec un DNS le serveur via l'adresse : <servername>.database.windows.net. Vous remarquerez que le nom du serveur a été généré automatiquement lors de la demande. Il n'est pas possible de modifier ce nom-là, cette politique a été mise en place par Microsoft pour éviter toute réclamation (par ex : un conflit sur le nom d'un serveur entre deux concurrents). Vous avez la possibilité de modifier votre mot de passe ou supprimer votre serveur depuis cette même page [Fig.2].

Pour créer une Database, il suffit de cliquer sur « create Database » et de nommer celle-ci en précisant sa taille (1Go, 10 Go, et 50 Go annoncé). On peut aussi récupérer la chaîne de connexion ODBC ou ADO.Net de chaque Database.



Dans le second onglet, on accède à toute la partie firewall. Pour renforcer encore la sécurité, Microsoft rajoute une couche de protection en mettant un filtre par adresse IP. En admettant qu'une tierce personne vous vole vos identifiants, si elle ne se connecte pas avec une IP comprise dans la plage d'adresse IP que vous avez définie, l'accès à la Database lui sera tout simplement refusé. La case « Allow Microsoft Services access to this server », autorise comme son nom l'indique l'accès des Services Microsoft au serveur [Fig.3].

Le management du serveur en ligne étant terminé, il ne vous



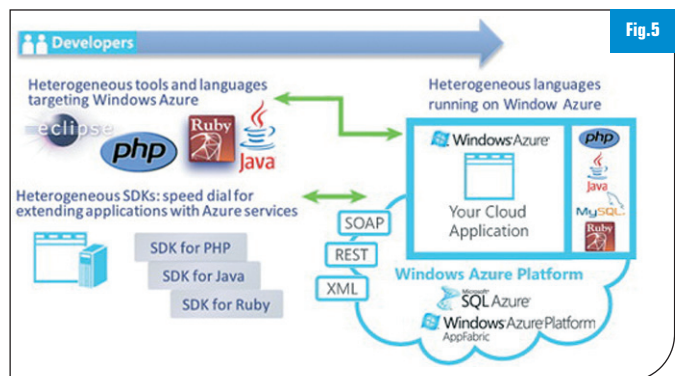
reste plus qu'à créer/importer votre base de données depuis SQL Server Management Studio. A noter qu'il vous faudra la version 2008 R2 qui a des implications dans le champ décisionnel (Business Intelligence) et de la montée en charge sur le Cloud. Une fois l'environnement lancé, vous devrez vous connecter via les identifiants fournis avec votre compte. **Rappel** : votre IP aura dû être au préalable inscrite dans la liste des plages IP autorisées dans la configuration firewall [Fig.4].

## L'interopérabilité, une volonté Microsoft

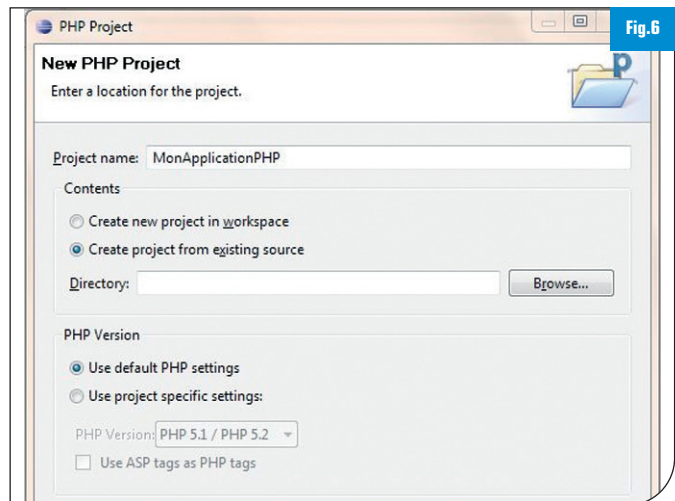
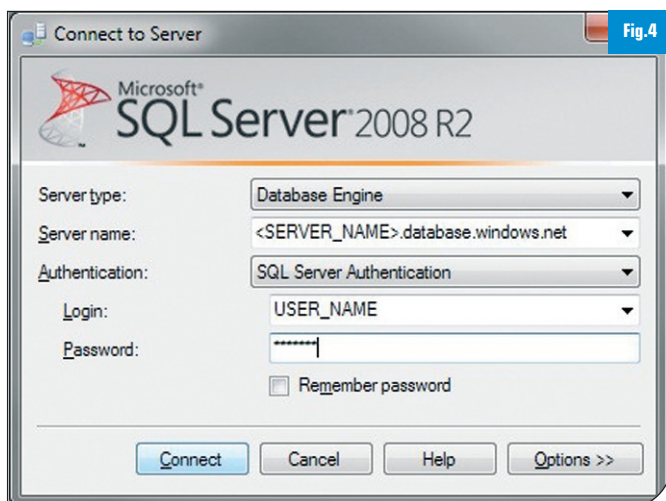
La plateforme Azure a été conçue pour pouvoir accueillir des services qui ne sont pas à l'origine du Framework .NET. On peut très bien retrouver d'autres langages hébergés sur la plateforme comme Java, PHP, Delphi ou encore Ruby. Cette interopérabilité est rendue possible par l'usage de protocoles internet standard, comme HTTP/HTTPS, REST ou encore XML. Au niveau du développement, cette souplesse dans le choix de l'utilisation du langage est facilitée par l'ajout de bibliothèques et d'outils développés par les communautés principales. Vous pouvez récupérer ces add-on via le site suivant : <http://www.microsoft.com/windowsazure/interop/>. [Fig.5]

### Une petite éclipse pour Visual Studio ?

Avec l'ouverture à d'autres langages de programmation, la question de l'utilisation d'autres IDE devient légitime. C'est le cas d'Eclipse qui devient alors une bonne alternative pour le développement d'applications qui n'appartiennent pas à la famille .NET. Nous allons voir comment on peut dès lors mettre en place un service en PHP et comment exploiter une base MySQL à partir de



Le schéma récapitulatif que l'on peut retrouver sur le site officiel





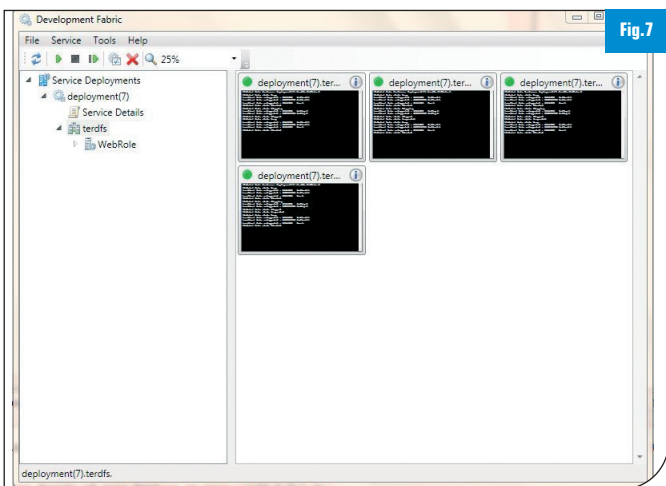


Fig.7

celui-ci. **Pré-requis** : il vous faut tout d'abord l'IDE Eclipse ainsi que les outils pour la plateforme Azure. Rendez-vous aux adresses respectives <http://www.eclipse.org/downloads/> et <http://www.windowsazure4e.org/download/>. Suivez le tutoriel pour finir l'installation.

Notre objectif maintenant va être de réutiliser une application PHP pour l'héberger « on the Cloud ». Cliquez dans la barre de menu sur File -> New -> PHP Project. Une nouvelle fenêtre s'ouvre, récupérez votre projet PHP et cliquez sur Finish [Fig.6].

À présent, faites un clic droit sur votre projet puis Windows Azure -> Convert to Windows Azure PHP Project. Nous choisirons pour la présentation l'option Windows Azure Data Storage & SQL Azure. Deux items sont alors présents dans le menu de gauche, MonApplicationPHP et MonApplicaionPHP\_WebRole. Le premier est une présentation de l'ensemble du service dans lequel on retrouve le fichier de configuration qui permet par exemple de modifier le nombre d'instances du rôle. Le second correspond à notre projet existant MonApplicationPHP qui inclut maintenant le namespace Microsoft ainsi que les librairies associées (méthodes utilisées pour le Storage et SQL Azure). Si vous désirez tester votre solution en local, il suffit de faire un clic droit sur MonApplicationPHP puis Windows Azure -> Run in Development Fabric. L'outil Development Fabric permet, lors du lancement, de mieux visualiser le service avec les instances créées pour un rôle [Fig.7]. Cette partie présente le développement PHP d'une solution héber-

geable dans le nuage. On va créer directement un nouveau Windows Azure Web project en choisissant toujours l'option Windows Azure Data Storage & SQL. Voyons les principales méthodes PHP qui serviront si on veut utiliser les Storage Blobs :

```
$olientStorage = new Microsoft_WindowsAzure_Storage_Blob();

//création d'un container
$result = $olientStorage->createContainer('MonContainer');
//suppression du container
$olientStorage->deleteContainer('MonContainer');
//ajout d'un blob au ContainerPrincipal
$olientStorage->putBlob('MonContainer', 'nomDuFuturBlob.ext', 'le/chemin/du/fichier');
//copie d'un blob
$olientStorage->copyBlob('MonContainer', 'BlobOriginal', 'MonContainer', 'BlobCopie');
//récupérer un blob pour ajouter le fichier dans un dossier
$olientStorage->getBlob('MonContainer', 'nomDuBlob', 'adresse/de/la/copie');
//rendre public un blob
$olientStorage->setContainerAcl('MonContainer', Microsoft_WindowsAzure_Storage_Blob::ACL_PUBLIC);
```

Le code suivant permet enfin de gérer toute la partie associée à la connexion en PHP à une base SQL Azure.

```
/*
Pour la connexion au serveur SQL Azure avec authentification, vous devez
modifier le paramètre $connectionInfo et affecter les attributs "UID",
"PWD" et "Database".
*/

$serverName = "NomduServer.database.windows.net";
$connectionInfo = array("UID"=>"VotreIdentifiant",
                        "PWD"=>"VotreMotDePasse",
                        "Database"=>"AdventureWorks");

$conn = sqlsrv_connect( 'top:', $serverName, $connectionInfo);
$query = "Requête à écrire ici";
//On assigne la valeur au paramètre littéral
$params = array ($value1,$value2);
sqlsrv_query($conn, $query, $params);

//Déconnexion
sqlsrv_close( $conn);
```

## Pour finir

Pour conclure cet article, nous pouvons dire que le produit Azure que propose Microsoft est une solution très avantageuse pour les entreprises qui désirent migrer sur le « Cloud ». Avec la possibilité de la montée en charge d'un service, les allocations de ressources à la volée en fonction des besoins et la facturation calculée sur l'utilisation réelle, Microsoft se présente comme un sérieux leader dans ce domaine en plein essor. Les développeurs ne sont pas en reste, de nombreux outils et SDK mis à disposition facilite le développement de services à destination du Cloud. L'environnement n'est plus restreint au framework .NET, les langages comme PHP y ont trouvé leur place. Pour résumer, la plateforme Azure marque un grand coup.

■ Damien Puig, *Développeur .NET*

■ Baptiste Lefrançois, *Développeur .NET*

### JUIN 2010

- Du 02 au 03 juin, Paris 75002, **Formation Intel Microsigma Programmation parallèle**, Techniques et Outils de Programmation Intel pour le multicore. <http://www.microsigma.fr/seminaires/InscriptionIntel1006.html>

- Le 3 juin 2010, Paris la Défense, Tour Ariane  
InterSystems organise un **atelier pratique et interactif** sur ses solutions : **Ensemble** et la base de données **Caché**. [http://www.intersystems.fr/page/fr/decouverte\\_technologies\\_intersystems.html](http://www.intersystems.fr/page/fr/decouverte_technologies_intersystems.html)

- Du 03 au 05 juin, Limoges, **Festival International du Webdesign 2010**  
<http://www.webdesign-festival.com/2010/spip.php?lang=fr>

- Le 8 juin 2010, Lille, Euratechnologies, **WygDay 2010** - l'édition Lilloise des Microsoft Days  
<http://wygday.wygwam.com/Inscription.aspx>

- Le 8 juin 2010, Microsoft France - Centre de Conférences 41 Quai du Président Roosevelt - 92130 Issy Les Moulineaux  
**Séminaires Business Intelligence en Libre-service avec Microsoft SQL Server 2008 R2 et PowerPivot**  
<http://www.finelog.fr>

- Du 11 au 12 juin 2010, Calais, Université du littoral côte d'Opale. **Les journées Perl 2010**  
<http://www.journeesperl.fr/>

### ETRANGER

- Du 07 au 11 juin, Etats-Unis, San Francisco, **conférence annuelle WWDC** (Worldwide Developers Conference) d'Apple.  
<http://developer.apple.com/wwdc>

- Du 09 au 12 juin, Allemagne, Berlin, **Linux Tag**, le Salon Européen de l'Open Source  
<http://www.linuxtag.org/2010/>

- Du 16 au 18 juin, Angleterre, Cambridge  
**Conférence Code Generation**, l'événement phare Européen sur la génération de code et le développement piloté par les modèles.  
<http://www.codegeneration.net/cg2010/>

### JUILLET 2010

- Du 03 au 10 juillet, Pologne, Varsovie, **Imagine Cup 2010**, [www.imaginecup.fr](http://www.imaginecup.fr)

# Intégrer les applications et exposer des services grâce à AppFabric

Les architectures de type « cloud » favorisent la réutilisation d'applications et de services : La création de nouvelles applications passe de plus en plus par l'intégration ou la réutilisation de fonctionnalités préexistantes (services ou applications), localisées ou non dans le nuage (hébergeur tiers, nuage ou système d'information privé), basées sur des technologies Microsoft ou non.

L'interconnexion de ces applications et services nécessite que les procédures d'identification et d'authentification soient interopérables et intégrables. Windows Azure AppFabric est un ensemble de fonctionnalités fourni par Windows Azure et permettant de répondre à ces besoins : rendre accessibles applications et services à travers Azure, fournir un système d'intégration de la sécurité. Pendant la phase Bêta d'Azure, Windows Azure AppFabric était connu sous le nom « .Net Services ».

On y trouve donc :

- Une couche de communication inter-application « **AppFabric Service Bus** » : il s'agit d'un bus de communication (basé sur WCF avec des bindings spécifiques) permettant l'interconnexion des applications, indépendamment de leur localisation (support du NAT, des firewalls, ...) ou de leur technologie
- Une couche d'interopérabilité de la sécurité « **AppFabric Access Control** », permettant à des applications distinctes d'échanger des informations d'identité sur les utilisateurs et aux utilisateurs d'une application d'accéder aux services d'une autre application.

Le nom « AppFabric » couvre aussi des fonctionnalités destinées à des serveurs Windows 2008 (Windows Server AppFabric) : cache de données réparti (ex Velocity), intégration d'application via Workflow Foundation et WCF (ex Dublin). Ces fonctionnalités ne seront pas couvertes dans cet article.

## Activation et configuration d'AppFabric

AppFabric est une fonctionnalité de Windows Azure. Pour l'utiliser, il vous faut donc au préalable un compte Azure. J'utiliserai ici le compte issu de mon abonnement MSDN (<http://msdn.microsoft.com/en-us/subscriptions/ee461076.aspx>).

L'activation et l'administration d'AppFabric se font via le site d'administration d'Azure [Fig.1].

Sélectionnez l'onglet « AppFabric » afin d'accéder à la liste des projets Azure éligibles à l'activation d'AppFabric [Fig.2 et 3].

Puis cliquez sur le lien « Add Service Namespace » afin de générer un URI unique qui sera nécessaire pour accéder aux fonctionnalités d'AppFabric [Fig.4].

J'utilise ici comme espace de nom « programmez-article-appfabric ». Bien entendu, vous devrez créer votre propre nom afin de garantir l'unicité de l'uri générée. La localisation du Datacenter Azure est importante car elle vous garantira les meilleures performances d'accès en fonction de la localisation géographique de vos utilisateurs. Le pack de connexions permet de spécifier le nombre de connexions simultanées possibles aux services exposés via AppFabric, ce qui aura un impact direct sur le coût d'utilisation de ce service (les informations sur les coûts d'Azure et d'AppFabric sont disponibles sur le site web de Microsoft).

Après avoir cliqué sur le bouton « Create », Azure procède à la création de l'espace de nom demandé et à la réservation des ressources correspondantes :

Service Namespace	Region	Created	Status
programmez-article-appfabric	Europe (North)	Sun, 02 May 2010 18:01:34 GMT	Activating...

Au bout de quelques dizaines de secondes, le namespace est actif :

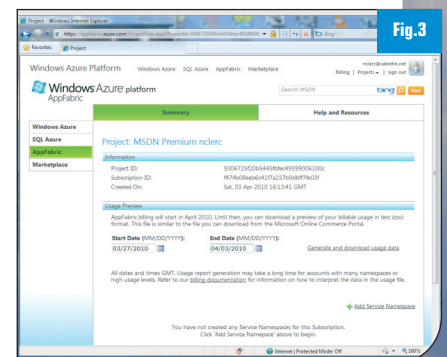
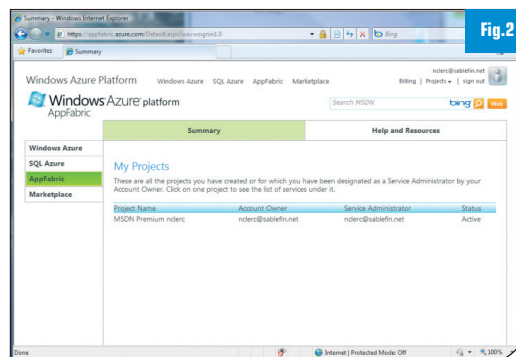
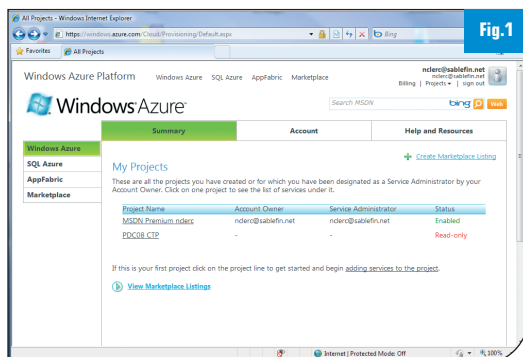
Service Namespace	Region	Created	Status
programmez-article-appfabric	Europe (North)	Sun, 02 May 2010 18:01:34 GMT	Active

Windows Azure a créé et activé de multiples url pour exposer les fonctionnalités d'AppFabric sur notre espace de nom : toutes ces url ont une racine commune : le nom du namespace créé.

- Service bus : <https://programmez-article-appfabric.servicebus.windows.net/>

- Access control : <https://programmez-article-appfabric.accesscontrol.windows.net>

La page d'administration de votre espace de nom permet de consulter la consommation des ressources qui y sont associées (nombre de connexions utilisées, pack actif, ...) [Fig.5].



## Configurer son poste de développement AppFabric

Une fois votre namespace activé, il vous faut maintenant configurer votre poste de développement :

- Installer Visual Studio 2010 Professional ou supérieur
- Télécharger et installer le SDK Windows Azure
- Télécharger et installer les outils de développement Azure pour Visual Studio
- Télécharger et installer le SDK Azure AppFabric

## Scénario des exemples

Au gré de mes péripéties sur les différentes conférences Microsoft, un scénario sympathique et tout à fait adapté à AppFabric s'est imposé : Le cas d'un ensemble hôtelier incluant restauration et boutiques.

- L'hôtel gère la réservation des chambres et la facturation du client
- Le restaurant à son propre logiciel de gestion, mais via le SI de l'hôtel il peut :
  - accepter des réservations
  - facturer sur la chambre du client
- La boutique de souvenirs a son propre logiciel de gestion mais permet de facturer sur la chambre du client.

Chacun de ces commerçants ne possède qu'une infrastructure réseau basique : connexion Adsl classique (pas de restriction en sortie, pas de port ouvert en entrée), logiciels métiers spécifiques, pas d'équipes ni de compétences informatiques avancées en interne. On obtient donc les échanges suivants : [Fig.6].

*HotelApplication* et *RestaurantApplication* exposeront les services correspondants aux fonctionnalités accessibles. L'implémentation de ces services restera classique : Nous ne nous concentrerons pas sur le fonctionnel (qui restera donc 'simulé' J ).

## Le service Bus : ou comment relayer des services

La fonctionnalité *ServiceBus* d'AppFabric fonctionne comme un intermédiaire de relais entre un service et un client. Traditionnellement, le client se connecte directement sur le point d'accès du service (*EndPoint*) pour dialoguer avec le service : [Fig.7].

Avec le bus de service d'AppFabric, l'accès aux services est relayé via *Windows Azure* : [Fig.8]. Le service exposé n'est pas accédé directement par les applications clientes, mais Windows Azure joue le rôle d'un intermédiaire de relais.

## Créer et exposer un service via le Service Bus

La manière de développer et de coder le service et le client ne change pas d'un service WCF classique.

Définition du contrat service via une interface :

```
[ServiceContract(Name = "IFacturationExterneContract",
    Namespace = "http://hotelapplication.sablefin.net/ServiceModel/FacturationExterne/")]
public interface IFacturationExterneContract
{
    [OperationContract]
    string CheckCustomer(long roomNumber, string customerName);

    // correspondant au jeton d'autorisation
    [OperationContract]
    string AddBillingLine(string customerToken, string description,
        decimal price, decimal vatRate, decimal vat);
}
```

Implémentation du service :

```
[ServiceBehavior(Name = "FacturationExterneService",
    Namespace = "http://hotelapplication.sablefin.com/ServiceModel/FacturationExterne/")]
public class FacturationExterneService : IFacturationExterneContract
{
    public string CheckCustomer(long roomNumber, string customerName) {...}

    public string AddBillingLine(string customerToken, string description,
        decimal price, decimal vatRate, decimal vat) {...}
}
```

L'exposition du service sera effectuée par le fichier de configuration de l'application. Il est aussi possible de le faire par programmation (voir le code source complet de l'application de démonstration). Configuration du service à exposer :

Ajouter dans le fichier App.config de l'application, dans la section <system.serviceModel/> :

- 1 • La configuration du binding *netTcpRelayBinding*.

**Service Namespace: programmez-article-appfabric**

**Manage**

Status: Active  
Delete: Delete Service Namespace

Management Key Name: owner  
Current Management Key: [redacted]  
Previous Management Key: [redacted]  
Generate New Key

**Service Bus**

Registry URL: https://programmez-article-appfabric.servicebus.windows.net/  
STS Endpoint: https://programmez-article-appfabric-sb.accesscontrol.windows.net/WRAPv0.9  
Management Endpoint: https://programmez-article-appfabric-sb.accesscontrol.windows.net/mgmt/  
Management STS Endpoint: https://programmez-article-appfabric-sb-mgmt.accesscontrol.windows.net/WRAPv0.9

Default Issuer Name: owner  
Default Issuer Key: [redacted]

**Access Control Service**

STS Endpoint: https://programmez-article-appfabric.accesscontrol.windows.net/WRAPv0.9  
Management Endpoint: https://programmez-article-appfabric.accesscontrol.windows.net/mgmt/  
Management STS Endpoint: https://programmez-article-appfabric-mgmt.accesscontrol.windows.net/WRAPv0.9

**Service Bus Connection Packs**

For more information regarding Connection Packs please review the AppFabric pricing information.

	Current	Pending
Connection Pack size:	5 Connections	No pending changes
Effective Date:	5/2/2010 6:01 PM GMT	

**Change Connection Pack size**  
This value will take effect in the next 24 hour billing cycle (starts 5/3/2010 12:00 AM GMT). Once a new Connection Pack size has taken effect, you must wait 7 days to change it again.

Change Connection Pack size to: 5 Connections [Change]  
Not eligible for changes until 5/9/2010 6:01 PM GMT.

**Information**

Project ID: [redacted]  
Created On: Sun, 02 May 2010 18:01:34 GMT  
Region: Europe (North)

Fig.5

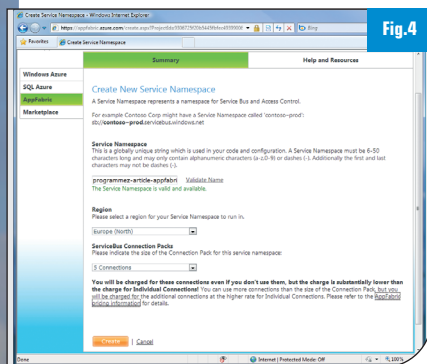


Fig.4

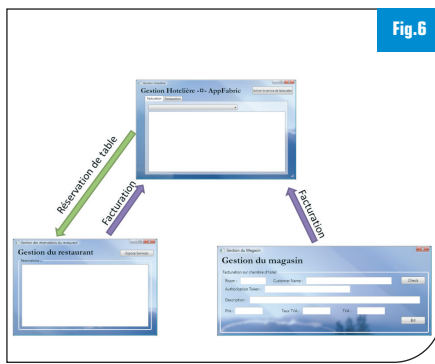


Fig.6

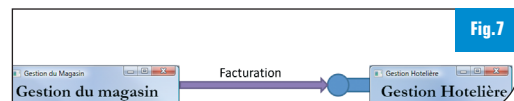


Fig.7

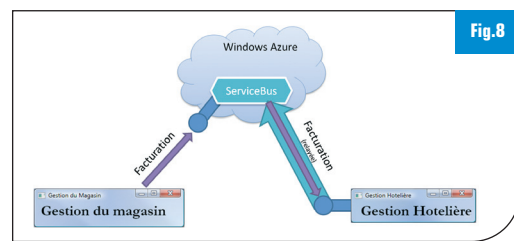


Fig.8



```
<netTcpRelayBinding>
  <binding name="defaultTcp">
    <security relayClientAuthenticationType="None" />
  </binding>
</netTcpRelayBinding>
```

2 • La configuration du behavior pour transmettre l'authentification *AppFabric* lors de l'enregistrement du service à relayer et de la création du point d'accès dans Azure.

```
<behavior name="sharedSecretClientCredentials">
  <transportClientEndpointBehavior credentialType="SharedSecret">
    <clientCredentials>
      <sharedSecret issuerName="owner" issuerSecret="KEY_APPFABRIC" />
    </clientCredentials>
  </transportClientEndpointBehavior>
</behavior>
```

Bien entendu, il faut remplacer « KEY\_APPFABRIC » par la clé chiffrée spécifique à votre espace de nom *AppFabric*.

3 • La définition du service utilisant le binding et le behavior

```
<service name="HotelApplication.Services.FacturationExterneService">
  <endpoint name="FacturationRelayEndpoint">
    <contract="HotelApplication.Services.IFacturationExterneContract">
      <binding="netTcpRelayBinding">
        <bindingConfiguration="tcpRelayWithoutClientAuthentication">
          <behaviorConfiguration="sharedSecretClientCredentials">
            <address="" />
          </behaviorConfiguration>
        </bindingConfiguration>
      </binding>
    </contract>
  </endpoint>
</service>
```

Exposition du service :

Il suffit ensuite de créer un host WCF pour le service et le démarrer.

```
Uri uriSbFacturation = ServiceBusEnvironment.CreateServiceUri("sb",
  "programmez-article-appfabric",
  "GestionHoteliere/Facturation");
ServiceHost hostServiceFacturation =
  new ServiceHost(typeof(Services.FacturationExterneService), uriSbFacturation);
hostServiceFacturation.Open();
```

*ServiceBusEnvironment.CreateServiceUri()* permet d'obtenir l'uri d'exposition du service : pour notre service, on obtient « sb://programmez-article-appfabric.servicebus.windows.net/GestionHoteliere/Facturation/ ».

L'ouverture du host (appel de la méthode *Open()*) déclenche les actions suivantes :

- 1 • Instanciation d'un binding *netTcpRelayBinding*
- 2 • Le binding se connecte sur *AppFabric* (en utilisant les informations d'authentification fournies dans le behavior)
- 3 • Le binding déclare à *AppFabric* le service à relayer
- 4 • *AppFabric* se met à l'écoute sur l'url d'exposition et maintient ouverte la connexion avec le service
- 5 • *AppFabric* relaiera les messages vers le service via la connexion maintenue ouverte.

Bien que notre application soit un service, l'exposition de ce service n'a nécessité qu'une connexion sortante, ce qui dans la plupart des cas ne posera aucun souci technique (le plus souvent, les firewalls sont largement plus tolérants en sortie qu'en entrée d'un réseau).

## Les différents Bindings disponibles :

Nous avons utilisé le *netTcpRelayBinding* qui permet d'exposer un service WCF de manière binaire en *Tcp*.

Bien entendu, les principaux bindings WCF ont leur équivalent pour le bus de message :

Binding WCF	Binding WCF avec ServiceBus
netTcpBinding	netTcpRelayBinding
	netEventRelayBinding
	netOneWayRelayBinding
basicHttpBinding	basicHttpRelayBinding
webHttpBinding	webHttpRelayBinding
wsHttpBinding	wsHttpRelayBinding

Le binding *netEventRelayBinding* est un peu particulier. Il permet d'avoir plusieurs applications à l'écoute sur un même endpoint. Son fonctionnement est unidirectionnel : tout client qui émettra un message en direction de cet endpoint verra son message distribué à chacun des services à l'écoute (on parle donc de distribution multicast) mais il ne recevra aucune donnée en retour (principe événementiel). Ce binding est dérivé du binding *netOneWayBinding* qui met en œuvre le même concept mais avec un seul service à l'écoute.

## L'optimisation des connexions

Dans le cas où le client et le service peuvent communiquer directement, le relais systématique des messages par le bus de service n'est pas optimal et introduira une latence inutile. Les bindings d'*AppFabric* basés sur le protocole *TCP* sont capables de détecter cette situation et de mettre en place un dialogue direct entre le service et le client. Ce comportement est configurable par l'intermédiaire de l'option *ConnectionMode* qui peut prendre 2 valeurs distinctes :

- **TcpRelayConnectionMode.Relayed** (valeur par défaut) : on force le routage des messages par le bus de service hébergé dans Azure (valeur par défaut).
- **TcpRelayConnectionMode.Hybrid** : détection et activation automatique de l'optimisation de routage local.

En mode hybride, si la connexion ne peut pas être obtenue (ou si la communication échoue), le binding bascule en mode *Relayed*.

L'état du mode de connexion est accessible par l'interface *IHybridConnectionStatus* :

```
IHybridConnectionStatus hybridConnectionStatus = channel.Get
Property<IHybridConnectionStatus>();
```

Cette interface expose l'événement "ConnectionStateChanged" qui maintiendra l'application informée des changements d'état du mode de connexion.

## L'interopérabilité

Les bindings basés sur *Tcp* sont conçus pour des services et des clients développés à l'aide des technologies .Net. Mais, à l'heure du *cloud computing* et de l'interopérabilité des services et applications, le bus de service d'*AppFabric* accepte les bindings basés sur *Soap* (*basicHttpRelayBinding*, *wsHttpRelayBinding*), mais aussi ceux utilisant *REST* (*webRelayHttpBinding*).

### Mise en œuvre de REST

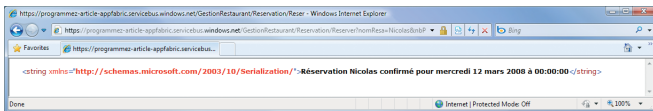
Dans nos applications, le service de réservation du restaurant est exposé en *REST* :

```
<service name="RestaurantApplication.Services.ReservationService">
  <behaviorConfiguration="withServiceMetadata">
    <endpoint name="RelayEndpoint">
      <contract="RestaurantApplication.Contracts.IRestaurantReservationContract">
        <binding="webHttpRelayBinding">
          <bindingConfiguration="default">
            <behaviorConfiguration="sharedSecretClientCredentials">
              <address="" />
            </behaviorConfiguration>
          </bindingConfiguration>
        </binding>
      </contract>
    </endpoint>
  </behaviorConfiguration>
</service>
```

Son contrat définit les caractéristiques spécifiques qui seront appliquées par le *webHttpBinding* (*REST*) : aspect de l'url, méthode d'accès, ...

```
[ServiceContract(Name = "IRestaurantReservationContract",
  Namespace = "http://restaurantapplication.sablefin.net/ServiceModel/Reservation/")]
public interface IRestaurantReservationContract
{
  [OperationContract] // déclaration d'un méthode de service
  [WebGet(UriTemplate = "/Reserver?nomRes={nomRes}&nbPersonnes={nbPersonnes}&dateEtHeure={dateEtHeure}")]
  string ReserverTable(string nomRes, long nbPersonnes, DateTime dateEtHeure, string remarque);
}
```

L'utilisation de *REST* permet d'accéder au service via un protocole facilement manipulable (voire directement à partir d'un navigateur).



Il est possible d'utiliser le formatage JSON pour les données retournées par le service, et permettre ainsi son accès en direct par des bibliothèques AJAX.

### Service Bus Message Buffer

Si l'utilisation de REST est encore trop compliquée, ou si les besoins de l'application sont réellement basiques, le bus de service supporte une fonctionnalité appelée « *MessageBuffer* ». Il s'agit d'une zone de stockage gérée par le bus de service qui permet à des applications de stocker et récupérer des messages. Le *MessageBuffer* fonctionne comme une file FIFO (First In First Out : les messages sont extraits dans l'ordre d'insertion : premier inséré premier extrait), avec la possibilité de verrouiller des messages dans la file. L'accès et le contrôle des *MessageBuffer* se font via l'api du *SDK AppFabric* mais aussi en utilisant le protocole REST, ce qui rend les *MessageBuffer* utilisables quelle que soit la technologie de l'application. L'utilisation en REST se fait à travers la combinaison d'un jeu d'url et des méthodes d'accès http : <http://serviceNamespace.servicebus.windows.net/{path}/{buffer}/{command}>

command	Méthode http	Action réalisée
{vide}	PUT	Création ou modification du buffer
	GET	Renvoie les règles de gestion du buffer
	DELETE	Suppression du buffer
/messages	POST	Crée un message et renvoie l'Uri d'accès au message
/messages/head	POST	Retourne le premier message disponible non verrouillé, et le verrouille. Les identifiants de verrou ( <i>LockId</i> ) et de message ( <i>MessageId</i> ) sont retournés.
	DELETE	Retourne le premier message disponible non verrouillé et le supprime du buffer.
/messages/{MessageId}	DELETE	Supprime le message dont l'identifiant est communiqué (le <i>LockId</i> est accepté).
/messages/{MessageId}/{LockId}	DELETE	Le message est déverrouillé.

Pour créer un *MessageBuffer* avec l'API il faut tout d'abord créer un behavior qui supportera les informations d'authentification :

```
TransportClientEndpointBehavior behavior = new TransportClientEndpointBehavior();
behavior.CredentialType = TransportClientCredentialType.SharedSecret;
behavior.Credentials.SharedSecret.IssuerName = "owner";
behavior.Credentials.SharedSecret.IssuerSecret = "VOTRE_ISSUER_KEY";
```

Ensuite, il faut définir les règles de fonctionnement du *MessageBuffer* (sécurité, durée de vie, visibilité, ...) :

```
MessageBufferPolicy policy = new MessageBufferPolicy();
policy.MaxMessageCount = 40;
policy.ExpiresAfter = TimeSpan.FromMinutes(2);
policy.Discoverability = DiscoverabilityPolicy.Public;
policy.OverflowPolicy = OverflowPolicy.RejectIncomingMessage;
policy.TransportProtection = TransportProtectionPolicy.None;
policy.Authorization = AuthorizationPolicy.NotRequired;
```

Ensuite, la classe *MessageBufferClient* va nous permettre de créer ce message buffer :

```
Uri uriMessageBuffer = new Uri("https://programmez-article-appfabric.servicebus.windows.net/Demo");
MessageBufferClient mbClient = MessageBufferClient.CreateMessageBuffer(behavior, uriMessageBuffer);
```

Pour supprimer le *MessageBuffer*, il suffira d'appeler la méthode *DeleteMessageBuffer()*. L'ajout d'un message dans le buffer passe par la création d'un message Soap qui sera envoyé au *MessageBuffer* :

```
mbClient.Send(
    Message.CreateMessage(
        MessageVersion.Soap12WSAddressing10,
        "urn:Message",
        "Ceci est mon message"
    )
);
```

La manipulation des messages présents dans le buffer fait appel à 4 méthodes :

- L'extraction d'un message utilisera la méthode *Retrieve()* :

```
Message msg = mbClient.Retrieve();
lblMessageRetrieved.Content = msg.GetBody<string>(); // "Ceci est mon Message"
```

- La méthode *PeekLock()* renvoie le premier message disponible, mais le laisse dans le buffer en le verrouillant (il ne sera plus disponible pour un *Retrieve()* ou un *PeekLock()* ).
- La méthode *Delete(msg)* supprime le message 'msg' du *MessageBuffer*.
- La méthode *ReleaseLock(msg)* libère le message qui était verrouillé (il redevient disponible pour un *Retrieve()*, un *PeekLock()* ou un *Delete(msg)* ) ;

### Le déploiement

En installant le *SDK AppFabric*, le fichier système machine.config est automatiquement mis à jour pour autoriser l'utilisation des bindings spécifiques dans le fichier de configuration des applications. Par contre, lors du déploiement de vos applications, ce SDK ne sera pas installé. Si vous utilisez les fichiers de configuration pour paramétrer vos services ou vos clients, il faudra - lors de la phase d'installation de votre application - mettre à jour le fichier *machine.config*. Pour cela, le SDK fournit un utilitaire « *RelayConfigurationInstaller.exe* » qui effectuera proprement cette tâche. Il se trouve dans le dossier du SDK : % ProgramFiles % \Windows Azure platform AppFabric SDK\IV1.0\Assemblies. Si besoin, il ne faut pas oublier de déployer l'assembly *Microsoft.ServiceBus.dll*.

### AppFabric Access Control

Le bus de services permet de relayer des messages et d'exposer des services. Il est fort probable que ces services doivent être sécurisés. De par la nature agnostique d'AppFabric vis-à-vis des technologies clientes utilisées, la sécurisation de ces échanges doit être interopérable.

C'est le rôle du contrôle d'accès fourni par AppFabric : intégrer et sécuriser des services et des clients quelle que soit la technologie utilisée pour les développer.

Les principales caractéristiques d'*AccessControl* sont :

- Accès sécurisé en HTTPS
- intégration native à ADFS (Active Directory Federation Services) : il s'agit des services d'intégration d'annuaires fournis par Windows
- mécanisme de sécurisation et d'authentification basé sur des clés symétriques et des signatures numériques SHA256
- règle de correspondance entre les différents éléments de sécurité (claim) fournis par un client ou attendus par un service
- support des protocoles WRAP (Web Resource Authorization Protocol) et SWT (Simple Web Token)

La mise en œuvre de l'*Access Control* fera l'objet d'un article à part entière dans un prochain numéro.

■ Nicolas Clerc - Microsoft Regional Director

Architecte et expert sur les technologies Microsoft

Blog : <http://www.sablefin.net>. Cofondateur de Tekigo, société de conseil et d'expertise technologique

# Exemple d'une migration ASP.net vers Azure

Lors de la Bewise Day Conference 2010 qui a eu lieu à Toulouse le 1er avril, le projet *nopCommerce Azure* a été officiellement lancé. Disponible sur CodePlex ce projet vous permet de créer votre site d'e-commerce en ligne tout en bénéficiant de tous les avantages de la plateforme Windows Azure. Vous le trouverez à cette adresse : <http://nopcommerceazure.codeplex.com>

L'application azurée se base sur le projet *nopCommerce*. Ce projet Open Source est disponible ici : <http://www.nopcommerce.com>. Il s'agit d'une application gratuite qui permet simplement et rapidement de créer son site web d'e-commerce. Nous allons passer en revue les différentes étapes indispensables pour rendre compatible cette application, développée en ASP.net, avec Windows Azure. *nopCommerce Azure* a servi de base au développement de sites web d'e-commerce comme la boutique de la célèbre marque de vêtement dia7 : [Fig.1].

La migration d'une application d'e-commerce vers la plateforme Windows Azure est très pertinente car ce type d'application possède des caractéristiques très particulières. Les sites d'e-commerce ont une fréquentation très fluctuante dans les périodes classiques en fonction de l'heure dans la journée, et du jour dans la semaine et ils doivent faire face à des pics de fréquentation potentiellement très importants sur les périodes de soldes ou après une diffusion de spot publicitaire télévisé. Migrer vers le cloud computing permet donc de dimensionner dynamiquement le nombre de serveurs qui vont héberger l'application en fonction de la charge effective, et ainsi avoir des coûts d'hébergement réduits par rapport à une offre traditionnelle.

## Architecture

Le projet *nopCommerce* est relativement bien construit d'un point de vue architecture. Les différentes couches et responsabilités sont bien définies et délimitées. Il n'est donc pas nécessaire de retravailler en profondeur l'architecture du projet existant.

On peut directement le modifier en ajoutant nos propres projets afin d'adjoindre les fonctionnalités propres à Windows Azure.

La première étape de notre migration est de rajouter un projet spécifique qui nous permettra de définir les différents rôles qui vont composer notre application Azure. Dans notre cas, cela est relativement simple, nous avons uniquement créé un Web Rôle qui nous permettra d'exposer le frontal web existant sous Windows Azure.

Pour cela, il faut ajouter un nouveau projet de type *Windows Cloud Service*, nommé *NopCommerceStore.Azure*. Il n'est pas ensuite nécessaire d'y mettre un rôle de type *ASP.net Web Role* car nous ne pouvons ajouter le projet *NopCommerceStore* qui constitue le frontal web. Une fois le projet créé, afficher le menu contextuel du dossier Roles et ajouter le projet web ASP.net présent dans la solution. La migration du frontal est d'ores et déjà terminée, nous



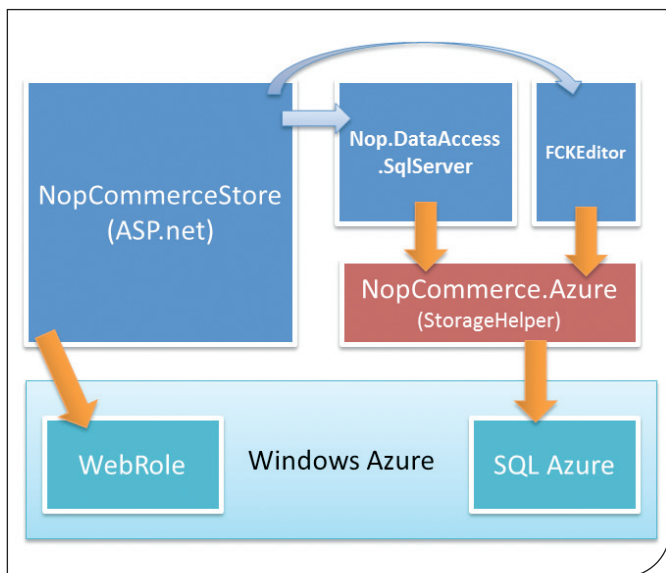
n'avons plus qu'à migrer le reste de l'application, en particulier le stockage de données et de fichiers.

Et c'est là que cela pourrait poser problème côté architecture puisque la persistance des données et la persistance des fichiers ne sont pas modulaires. Pour continuer notre migration nous allons donc voir comment modifier les projets existants afin d'utiliser les différents services proposés par Windows Azure.

## L'accès aux données

*nopCommerce* utilise SQL Server pour stocker ses données. La migration vers SQL Azure est donc assez simple. Côté outillage, vous pouvez gérer votre base de données SQL Azure avec SQL Server Management Studio 2008 R2, c'est-à-dire le même outil de gestion que vos bases de données SQL Server 2005/2008. Côté technique, il n'est pas nécessaire de modifier la couche d'accès aux données car l'accès à une base de données SQL Azure se fait de la même manière que SQL Server. Les seules modifications qui sont nécessaires concernent les scripts T-SQL car SQL Azure supporte un sous-ensemble des instructions supportées par SQL Server. Par exemple le type text et le type image ne sont pas supportés. Il faut utiliser *varchar(max)* et *varbinary(max)* à la place de ces types. Afin de faciliter l'installation de l'application il a donc été nécessaire de modifier le script de création de la base de données





nopCommerce\_createDatabase.sql afin de remplacer ces instructions et opérateurs T-SQL non supportés.

## Stockage de fichiers et d'images

Le stockage de fichiers et des images (utilisées pour les photos de produits par exemple) a également dû être redéveloppé. Afin de bénéficier d'une bonne montée en charge, il est en effet nécessaire de stocker ce type d'informations non plus sous forme de simples fichiers mais de stocker ces informations en utilisant le service de stockage de blobs proposé par Windows Azure.

Le service de blob permet de stocker des données brutes telles que des fichiers. Il permet en plus de les exposer via un service de Content Delivery Network (CDN) afin d'optimiser les vitesses de transfert en mettant en cache les données sur différents serveurs répartis idéalement sur la planète. Ainsi un utilisateur américain n'aura pas besoin de télécharger vos fichiers couramment téléchargés depuis votre serveur européen sur lequel vous l'hébergez, mais pourra directement récupérer une version en cache téléchargée par un serveur situé dans un datacenter américain.

Pour stocker les fichiers et les images dans des blobs, nous avons donc créé un projet de type Class Library afin d'ajouter une classe StorageHelper.cs qui permet d'effectuer toutes les opérations d'accès aux fichiers comme l'upload, la lecture, ou l'énumération de fichiers en les remplaçant par des accès à des blobs. La manipulation de blobs est très simple puisqu'il suffit d'utiliser le SDK Windows Azure fourni par Microsoft pour gérer ce type de stockage. Exemple pour uploader des images :

```

/// <summary>
/// Upload a file from a media with specified name
/// </summary>
public static String UploadFile(Bitmap media, string
imageFormat, String containerName, string uniqueBlobName)
{
    if (!storageInitialized)
        InitializeNopStorage();

    CheckIllegalArguments(containerName, uniqueBlob
Name);

```

```

// Get reference
var cloudBlockBlob = GetCloudBlockBlob(container
Name, uniqueBlobName);

/* Image processing removed */

try
{
    using (MemoryStream stream = new MemoryStream())
    {
        media.Save(stream, codecInfo, ep);
        stream.Position = 0;
        cloudBlockBlob.UploadFromStream(stream, Options);
    }

    return uniqueBlobName;
}
catch (WebException webEx)
{
    throw new WebException(
        "Error while trying to Upload Http File on
Azure Storage", webEx);
}
}

```

## Modification de fckEditor

nopCommerce utilise par défaut le composant fckEditor pour permettre à l'utilisateur de saisir du contenu riche et de gérer ses images en lui donnant la possibilité d'uploader ses propres images et également de pointer vers du contenu existant.

Et c'est sur ce dernier point qu'il faut intervenir puisque nous devons modifier le comportement de ce composant qui est normalement conçu pour uploader les fichiers sur un répertoire physique sur le serveur web, et non pour uploader des données vers des blobs hébergées sur Windows Azure.

Il faut donc télécharger le composant FCKEditor accompagné de ses sources pour modifier le projet FredCL.FCKEditorV2.vs2005. Le fichier Connector.cs a été modifié afin de remplacer les accès au système de fichiers par des appels à la classe StorageHelper.cs créée précédemment.

La modification n'est donc pas importante en terme de charge mais il faut bien penser à migrer l'ensemble des dépendances de vos projets existants pour que ceux-ci supportent Windows Azure à leur tour.

## Conclusion

La migration d'un projet ASP.net existant vers Windows Azure n'est pas très complexe et est facilitée grâce à l'outillage fourni par Visual Studio. Les points importants à prendre en compte concernent l'exposition de l'application en termes de rôles (web rôles et worker rôles), et les différentes stratégies de stockage (base de données et blobs pour les fichiers). Une fois ces éléments migrés, votre application est prête à être déployée sur Windows Azure !

### ■ Patrice Lamarche

Responsable du domaine Solution Langages et Framework à Bewise.

Membre du Microsoft Windows Azure Circle.

Blog : <http://patricelamarche.net> Site : <http://www.bewise.fr>

# DÉVELOPPEZ VOTRE SAVOIR-FAIRE



Langage et code, développement web,  
carrières et métier :  
Programmez !, c'est votre outil  
de veille technologique.

Pour votre développement personnel et professionnel,  
abonnez-vous à Programmez !

## Choisissez votre formule

- Abonnement 1 an au magazine : 49 €  
(au lieu de 65,45 € tarif au numéro) *Tarif France métropolitaine*
- Abonnement Intégral : 1 an au magazine + archives  
sur Internet et PDF : 59 € *Tarif France métropolitaine*
- Abonnement PDF / 1 an : 30 € - *Tarif unique*  
Inscription et paiement **exclusivement en ligne**  
[www.programmez.com](http://www.programmez.com)
- Abonnement Etudiant : 1 an au magazine : 39 €  
(au lieu de 65,45 € tarif au numéro) *Offre France métropolitaine*

11 numéros par an : 49 €\*

**Economisez 16,45 €\***

\*Tarif France métropolitaine

## + Abonnement INTÉGRAL

**ACCÈS ILLIMITÉ aux ARCHIVES du MAGAZINE pour 0,84€ par mois !**

Cette option est réservée aux abonnés pour 1 an au magazine,  
quel que soit le type d'abonnement (Standard, Numérique, Etudiant).  
Le prix de leur abonnement normal est majoré de 10 € (prix identique

pour toutes zones géographiques). Pendant la durée de leur abonnement,  
ils ont ainsi accès, en supplément, à tous les anciens numéros et articles/  
dossiers parus.

**OUI, je m'abonne** Vous pouvez vous abonner en ligne et trouver tous les tarifs [www.programmez.com](http://www.programmez.com)

### PROGRAMMEZ

- ☐ Abonnement 1 an au magazine : 49 € (au lieu de 65,45 € tarif au numéro) *Tarif France métropolitaine*
- ☐ Abonnement Intégral : 1 an au magazine + archives sur Internet et PDF : 59 € *Tarif France métropolitaine*
- ☐ Abonnement Etudiant : 1 an au magazine : 39 € (joindre copie carte étudiant) *Offre France métropolitaine*

☐ M. ☐ Mme ☐ Mlle Entreprise : ..... Fonction : .....

Nom : ..... Prénom : .....

Adresse : .....

Code postal : ..... Ville : .....

Tél : ..... E-mail : .....

☐ Je joins mon règlement par chèque à l'ordre de Programmez ! ☐ Je souhaite régler à réception de facture

A remplir et retourner sous enveloppe affranchie à :  
Programmez ! - Service Abonnements - 22 rue René Boulanger - 75472 Paris Cedex 10.  
[abonnements.programmez@groupe-gli.com](mailto:abonnements.programmez@groupe-gli.com)

**Offre limitée,**  
valable jusqu'au  
31 mai 2010

Le renvoi du présent bulletin implique  
pour le souscripteur l'acceptation  
pleine et entière de toutes les  
conditions de vente de cette offre.  
Conformément à la loi Informatique et  
Libertés du 05/01/78, vous disposez  
d'un droit d'accès et de rectification  
aux données vous concernant.  
Par notre intermédiaire, vous pouvez  
être amené à recevoir des propositions  
d'autres sociétés ou associations.  
Si vous ne le souhaitez pas, il vous  
suffit de nous écrire en nous précisant  
toutes vos coordonnées.

Le magazine du développement  
**PROgrammez !**  
[www.programmez.com](http://www.programmez.com)

# Choisir sa forge pour ses projets libres

Forges, compilateurs, environnements de développement, outils de gestion de versions, tous les moyens techniques existent pour qui souhaite se lancer dans un projet libre. Comme il est d'usage, pour le développeur, le premier pas est aussi le plus difficile : se décider à publier son code.

Le mode de développement collaboratif que symbolise le logiciel libre est devenu la norme en quelques années. Pratiquement tous les éditeurs importants proposent des sites de collaboration, et les forges qui hébergent les projets libres n'ont jamais été aussi nombreuses. SourceForge, la plus ancienne, a référencé plus de 230 000 projets de tous types dans son histoire, et un peu plus de 150 000 actuellement. Depuis SourceForge, une vingtaine de forges se sont constituées, généralistes ou spécialisées sur un domaine donné, comme les outils de développement. Parmi les plus connues, citons Github, GNU Savannah, Googlecode ou encore Codeplex, lancée par Microsoft. Github, l'une des plus récentes, propose des comptes payants pour les projets propriétaires, et gratuits pour les projets libres.

Le mode de développement à base de forge s'est répandu très largement dans l'industrie, et on le retrouve au centre des activités des grands consortiums tels que les fondations Apache et Eclipse ou encore OW2. Bull, l'un des fondateurs d'OW2, utilise d'ailleurs cette approche pour mettre en relation ses ingénieurs et ses clients avec son offre Novaforge. Le principe de la forge répond à un besoin de visibilité globale : l'ensemble des contributeurs forme une commu-



*SourceForge a été lancée en 1999, et référence plus de 160 000 projets.*

nauté extrêmement dispersée, et il était nécessaire de disposer d'un point central où le maximum de projets était référencé. La forge intègre de manière cohérente tous les outils nécessaires aux développeurs, au lieu d'avoir à les installer séparément et pour chaque contributeur. Elle n'est toutefois pas seulement réservée aux développeurs. D'autres acteurs peuvent également participer, pour la traduction, les tests ou la documentation par exemple.

Les fonctions les plus courantes d'une forge incluent bien entendu l'hébergement du code du projet, mais aussi la gestion de versions comme via Subversion, CVS, Bazaar ou Git par exemple, le suivi des bugs et une liste de diffusion par projet facilitant les échanges. Parfois, elles proposent des fonctions de travail collaboratif, comme la revue de code, de suivi des tâches ou des services de traduction.

Les forges sont construites sur des logiciels, pas toujours libres d'ailleurs, qui regroupent ces fonctions. Ainsi SourceForge s'appuie sur le logiciel propriétaire TeamForge appartenant à Geeknet, société qui assure également le fonctionnement de Freshmeat, de Slashdot et du site de commerce ThinkGeek. Beaucoup de logiciels libres de ce type sont toutefois accessibles. Les grandes forges généralistes sont cependant deve-

nues de tels mastodontes qu'il devient difficile d'y trouver son chemin.

## Pour démarrer : publier!

« Les forges telles qu'elles ont commencé dans les années 90 tendent à disparaître, explique Loïc Dachary, l'un des pionniers du logiciel libre en France. Elles ont été le symbole d'un mode de développement, mais il n'y a plus de raisons d'être complètement centralisé. Tous les développeurs ont désormais une copie intégrale de la forge. La tendance actuelle est d'avoir un site par projet, ce qui est souhaitable, d'abord parce que cela évite de dépendre d'une ressource centralisée, mais aussi parce que cela participe d'une amélioration du processus de développement. Ce n'est pas la forge qui donne de la visibilité aux projets, c'est l'inverse. Pour trouver les projets, les moteurs de recherche sont efficaces. »

La publication est la première et plus importante étape pour qui souhaite se lancer dans un projet libre. C'est aussi la plus difficile, le plus souvent. Elle implique de dépasser une timidité très fréquente, et d'oser publier son code, même s'il n'a rien de glorieux. Selon Loïc Dachary, le principal obstacle, c'est de se convaincre de publier. Il faut copier ce que l'on a écrit dans un endroit public. C'est souvent un problème pour les entreprises, qui ont des difficultés à estimer ce qui, dans leur parc applicatif, peut bénéficier de ce mode de développement et ce qui doit demeurer dans l'entreprise en raison de l'avantage concurrentiel qu'il représente.

« Les gens sont réticents à la publication, constate Eric Leblond, directeur technique d'Edenwall. Il y a l'exemple célèbre de Linus Torvalds lui-même, à qui l'on demandait de montrer un morceau de code lors du procès SCO



*GitHub, l'une des plus récentes (2008) affiche l'une des plus fortes croissances.*



et qui disait qu'il en avait honte, mais qu'il l'avait quand même publié. Il y a deux attitudes, l'une relevant du blocage, de la timidité, de l'angoisse, et l'autre partant du principe que d'autres l'auront sûrement déjà fait. Mais il faut contribuer dès que possible, même sur de petites choses. Particulièrement sur des petites choses au début, car ce sont les petits projets qui sont les plus accessibles, et sur lesquels il y a proportionnellement le plus de contribu-

tions. » Quoi qu'il en soit, la première question est de savoir quel statut viser, selon que l'on veut démarrer son propre projet ou contribuer à un projet existant.

« Les deux démarches sont très différentes, explique Loïc Dachary. Pour ce qui est de la contribution, la première chose est de savoir à quoi contribuer. On recherche un projet, sur une forge ou ailleurs, et là plusieurs attitudes sont possibles. En fait, c'est tout l'éventail des attitudes

d'une personne qui souhaite intégrer un groupe existant : on essaie de se rendre utile, dans le but d'être accepté par les autres. Si l'on est solitaire, on écoute les discussions, les forums d'échanges, et l'on détermine une correction que les gens réclament, un problème à corriger. À partir de là, on s'arrange pour trouver une correction, et on la publie. »

■ Philippe Davy

## Les principales forges généralistes

Forge	Site	Principaux projets	Nombre de projets	Utilisateurs	Fonctions	Gestion de versions
ActiveState Firefly	<a href="http://firefly.activestate.com">http://firefly.activestate.com</a>	Drupal for Komodo, PyPM	670	-	Hébergement du code, suivi des bugs, wiki, forum, référentiel privé, gestion d'équipe	SVN
Alioth	<a href="http://alioth.debian.org/">http://alioth.debian.org/</a>	SANE	800	8 500	Hébergement du code, suivi des bugs, référentiel privé	CVS, SVN, Bazaar, Arch, Git, Mercurial
Assembla	<a href="http://www.assembla.com/">http://www.assembla.com/</a>	Drupal for Komodo, PyPM,	60 000	170 000	Hébergement du code, suivi des bugs, wiki, forum, référentiel privé, gestion d'équipe	SVN, Git, Mercurial
BerliOS	<a href="http://www.berlios.de/">http://www.berlios.de/</a>	Git, avidemux, Haiku, J2ME Polish, Code::Blocks IDE	5 300	43 000	Hébergement, suivi des bugs, wiki, forum	CVS, SVN
Bitbucket	<a href="http://bitbucket.org/">http://bitbucket.org/</a>	TortoiseHG, Aldrin, Pylons	1 300	35 000	Hébergement (1 Go) et analyse du code, suivi des bugs, wiki, référentiel privé, gestion d'équipe	Mercurial
CodePlex	<a href="http://www.codeplex.com/">http://www.codeplex.com/</a>	Ajax CTK, Silverlight Toolkit, IronPython	9 200	150 000	Hébergement du code, suivi des bugs, wiki, forum	CVS, Team Foundation Server
Fedora Hosted	<a href="https://fedorahosted.org/">https://fedorahosted.org/</a>	Fedora, LHCP	600	-	Hébergement et analyse du code, suivi des bugs, wiki, gestion d'équipe	SVN, Git
GitHub	<a href="http://github.com/">http://github.com/</a>	Ruby on rails, IronRuby, Perl	90 000	200 000	Hébergement (300 Mo) et analyse du code, suivi des bugs, wiki, référentiel privé	Git
Gitorius	<a href="http://gitorious.org/">http://gitorious.org/</a>	openSuse, Qt, FreeBSD, Amarok	-	-	Hébergement et analyse du code, wiki, gestion des équipes	Git
GNA	<a href="http://gna.org/">http://gna.org/</a>	Bamboo, BeWos	1 300	13 700	Hébergement, suivi des bugs, supervision projets	CVS, SVN, Arch
GNU Savannah	<a href="http://savannah.gnu.org/">http://savannah.gnu.org/</a>	Projets FSF, Emacs, GIFT, Classpath Extensions	3 000	47 000	Hébergement, suivi des bugs, forum, gestion des équipes	CVS, SVN, Bazaar, Arch, Git, Mercurial
Google Code	<a href="http://code.google.com">http://code.google.com</a>	Google Gears, Android, Chromium	250 000	-	Hébergement et analyse du code, suivi des bugs, wiki, traduction	CVS, Mercurial
JavaForge	<a href="http://javaforge.com/">http://javaforge.com/</a>	Projets Java	-	-	Hébergement du code, suivi des bugs, wiki, forum, gestion d'équipes, intégration Eclipse et Netbeans	CVS, SVN, Git, Mercurial
Knowledgeforge	<a href="http://knowledgeforge.net/">http://knowledgeforge.net/</a>	Kforge, HeWit	190	900	Hébergement et analyse du code, suivi des bugs, wiki, forum, référentiel privé, gestion des équipes	CVS, Git, Mercurial
Launchpad	<a href="https://launchpad.net/">https://launchpad.net/</a>	Ubuntu, MySQL, Inkscape, Zope 3	15 000	900 000	Hébergement et analyse du code, suivi des bugs, forum, référentiel privé, gestion des équipes	Bazaar
Origo	<a href="http://www.origo.ethz.ch">http://www.origo.ethz.ch</a>	Tft, Translation Lifecycle Manager	2 000	-	Hébergement, suivi des bugs, wiki, forum, intégration Eclipse	SVN
OSOR.eu	<a href="http://www.osor.eu/">http://www.osor.eu/</a>	Applications par et à l'usage des administrations européennes	2 100	-	Hébergement, suivi des bugs, wiki, forum	SVN
SourceForge	<a href="http://sf.net/">http://sf.net/</a>	Lame, Audacity, 7-Zip, Emule, Filezilla, ffdshow	160 000	2 600 000	Hébergement, métriques projets, suivi des bugs, forum	CVS, SVN, Bazaar, Git, Mercurial
Tigris.org	<a href="http://www.tigris.org/">http://www.tigris.org/</a>	ArgoUML, Subversion, sstree, frameworkx	1 500	135 000	Hébergement, suivi des bugs	CVS, SVN

# Entre crise et restructuration, un marché qui se redresse

**HAYS**  
Recruiting experts  
in Information Technology

Cette étude est  
réalisée par Hays,  
Cabinet Conseil en  
recrutement,  
en partenariat avec  
LesJeudis.com  
et l'école  
Télécom Sud Paris.

L'année 2008-2009 a bousculé le monde du travail.  
La conjoncture a eu un impact important sur les recrutements  
et les politiques de rémunération.

## 40 % de postes en moins en 2009

Le secteur Informatique & Télécoms, comme tous les secteurs de l'économie, a été victime de la crise. Le nombre de postes recensés par l'APEC a fortement diminué : 40 % de postes en moins en 2009 ; ce qui élèverait en fin d'année le taux de chô-

mage des informaticiens à 6 %.

En même temps, une restructuration du marché se poursuit de façon inexorable : cette année, la création d'emplois en offshore chez les grandes SSII françaises dépasse celle de l'hexagone. Cependant, le cabinet Hays travaille peu avec des SSII et se positionne plutôt comme accompa-

gnateur de jeunes sociétés et de clients finaux. Malgré la conjoncture, nous observons de fortes demandes auprès de nos clients (éditeurs français) à la pointe de l'innovation. Ce sont ces sociétés qui seront les créateurs d'emplois de demain. Nous constatons également que les Ingénieurs informatiques et télécoms en France bénéficient d'une des meilleures formations dans le monde. Ainsi, pourront-ils faire face à la concurrence internationale du marché de l'emploi et des restructurations du marché. Ces jeunes Ingénieurs sont nos futurs entrepreneurs et donc les futurs créateurs d'emplois.

## AVIS D'EXPERT

### Les profils les plus demandés

Les entreprises se développant sur des marchés innovants à forte valeur technologique ont continué à recruter et ont su maintenir une croissance à deux chiffres en 2009.

**L**es secteurs tels qu'internet, le multimédia, le jeu vidéo et la sécurité informatique ont le vent en poupe malgré la conjoncture.

#### Les sociétés de services ont perdu de leur attractivité au détriment des clients finaux

L'année 2009 a pris un nouveau virage sur le marché de l'emploi IT tant du côté des candidats que du côté des entreprises. Les sociétés de services ont perdu de leur attractivité au détriment des clients finaux tandis que les Ingénieurs confirmés par exemple, souhaitent aujourd'hui s'orienter vers un poste en interne pour y trouver des challenges, une stabilité et des perspectives d'évolution. De leur côté, les entreprises recherchent toujours des compétences de plus en plus pointues en maintenant une politique de rémunération d'embauche peu flexible.

Compétences les plus recherchées :

- maîtrise de Spring, Struts, Hibernate. pour les Ingénieurs Java/JEE
- maîtrise de .Net 3.5, SQL Server 2005/2008 pour les ingénieurs .Net
- maîtrise de Zend ou Symfony pour les développeurs PHP.

C'est pourquoi les formations aux nouvelles technologies (via des certifications, formations internes, DIF, CIF...) sont devenues très importantes. Les mutations technologiques sont plus fréquentes et se former régulièrement permet de rester attractif pour les entreprises en croissance. Un candidat, qu'il soit en recherche d'emploi ou en poste, se doit de se former rapidement aux nouvelles technologies qui arrivent sur le marché. Un exemple concret cette année est la hausse de la demande de compétences sur Sharepoint et Moos pour les développeurs .Net.

#### Les salaires

Concernant les rémunérations, elles n'ont en général pas augmenté en 2009. Les salaires proposés par nos clients aujourd'hui correspondent plutôt aux salaires de 2008 et sont dans certains cas revus à la baisse. Pour l'année 2010, la tendance est à la stabilité, sauf pour les profils très spécifiques aux compétences rares.

■ **Corentin Siffroi**  
Consultant Senior, Hays



■ **Candice Ouaknine**  
Consultante, Hays



#### 2010, une année encore difficile

Entre crise et restructuration, les salaires des Informaticiens ont stagné en 2009. Ce ralentissement a été ressenti comme un choc brutal auprès des informaticiens habitués aux années fastes de 2004 à 2008. Toutefois, nous avons pu noter quelques hausses de salaire cette année, en particulier pour des profils rares et pointus. En 2010, le marché de l'emploi en Informatique & Télécoms restera en berne malgré la flexibilité du modèle de sous-traitance, qui pourra créer des emplois aussi rapidement qu'elle les a détruits en 2009. Si la reprise budgétaire ne se manifeste pas rapidement, 2010 pourrait être encore une année difficile tant pour les salaires que pour la création d'emplois.

■ **Anthony Collins** ►  
Directeur, Hays

■ **Yann Bell**  
Manager, Hays



“ Se former régulièrement permet de rester attractif ”





# Web et Open source :

## Forte demande, mais pénurie des candidats qualifiés et disponibles

Deux classes de métiers résistent bien à cette période de récession. Les métiers liés au marketing Online plutôt "Front office" tels que les Webdesigners, les Community Manager et les Webmarketers sont très recherchés.

### Salaires ETUDE ET DEVELOPPEMENT

	Jeune Dip	1 - 2 ans	2 - 5 ans	5 - 10 ans	10 ans et plus
Développeur	26 / 28 / 30	28 / 30 / 32	30 / 33 / 35	35 / 40 / 45	45 / 50 / 55
Ingénieur de développement	32 / 34 / 35	28 / 30 / 32	38 / 41 / 44	44 / 50 / 55	50 / 55 / 60
Ingénieur tests & validations	30 / 32 / 33	28 / 30 / 32	36 / 39 / 42	40 / 45 / 50	45 / 50 / 55
Chef de projet technique	NS	NS	40 / 43 / 45	45 / 53 / 60	55 / 60 / 65
Architecte logiciel	NS	NS	NS	45 / 53 / 60	55 / 60 / 65
Chef de projet fonctionnel	NS	NS	NS	55 / 65 / 75	65 / 74 / 80

Rémunération en €. (Source Hays) .../.../.../ Le 1er chiffre indique le niveau haut des rémunérations pratiquées, le troisième chiffre le niveau bas. NS : non significatif.

NB : l'échantillon interrogé est majoritairement basé sur Paris-RP. Les salaires en région peuvent être inférieurs de 10% ou plus.

Réelle conséquence du succès du web 2.0, des communautés du référencement et de l'affiliation, les entreprises n'hésitent pas à se munir de ces "nouveaux" experts.

### Compétences en logiciels libres

Les autres métiers sont plutôt "back office" et sont liés aux technologies Open Source, tels que les Architectes web, Chefs de projets techniques et Développeurs sur des environnements Linux, Apache, MySQL, PHP. En effet, une des conséquences des réductions budgétaires en matière de systèmes d'information est le recours croissant aux logiciels libres.

L'OS est partout et permet aux entreprises de toutes tailles d'accéder à des technologies innovantes à moindre coût.

La forte demande du marché se concentre surtout sur les technologies PHP, plutôt orientées objet.

### Autre tendance : le framework.

La grande majorité des entreprises démarrant des projets informatiques s'accorde sur le fait qu'il est indispensable d'avoir un framework de développement, permettant de normaliser et d'accélérer les développements. Les compétences sur des frameworks tels que Zend ou Symfony sont très demandées et appréciées. Mais nous rencontrons sur ces profils une vraie guerre des talents, avec une demande croissante des entreprises et une pénurie des candidats qualifiés et disponibles.

## L'expérience d'un ingénieur chercheur d'emploi : On demande des profils "à tout faire"

« Ingénieur en informatique depuis 12 ans et suite à des restructurations dans le secteur des Télécoms, j'ai été amené à rechercher un emploi 3 mois avant la date prévue de mon licenciement. Ces trois mois m'ont permis de me rendre compte de la situation du marché de l'emploi dans le secteur de l'informatique.

### Il faut faire des concessions

Lorsque j'ai débuté ma recherche, je posais mes conditions et mes exigences quant au poste que je désirais. Le marché étant assez morose durant cette période, je me suis aperçu que je risquais de me retrouver au chômage si je ne faisais pas des concessions sur certaines conditions. Aussi, j'ai signé pour un poste qui ne correspondait pas à l'ensemble de mes attentes mais qui présentait l'avantage d'être un poste stable et d'être situé à proximité de mon domicile. Depuis

septembre 2009, j'ai pu constater une reprise de la demande dans le secteur de l'informatique et les profils de Développeurs et d'Administrateurs sont très demandés et

“ L'employeur ne diminue pas les salaires des profils expérimentés mais il leur demande plus de compétences. ”

rare. Personnellement j'ai été contacté à plusieurs reprises. Concernant les salaires, le constat est clair : il n'y a pas de réelle augmentation, ni de réelle diminution. Il convient cependant de modérer ce propos ; nous avons une tendance pernicieuse à la cumulation des fonctions au sein d'un même poste. De plus, on demande des profils « à tout

faire » : rédaction des spécifications fonctionnelles et techniques, planification des tâches, développement et administration systèmes et réseaux. C'est le profil du poste que j'occupe actuellement, à salaire égal. Autrement dit, l'employeur ne diminue pas les salaires des profils expérimentés mais il leur demande plus de compétences. »

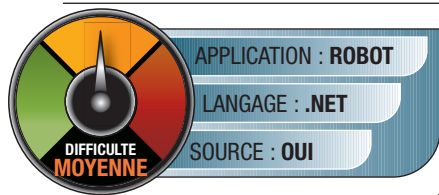
Candidat rencontré par Hays en 2009



# Microsoft Robotics Developer Studio : La robotique étend ses frontières

2<sup>e</sup> partie

Après les articles des numéros 120 et 121 de Frédéric Mazué sur le cœur de Microsoft Robotics Developer Studio, CCR et DSS, et après celui du numéro 128 sur la découverte du Visual Programming Language et de l'environnement Robotics Studio, intéressons-nous maintenant à une application tout à fait concrète de la robotique dans la vie de tous les jours, la domotique.



L'automatisation de la maison est un vaste sujet qui couvre aussi bien le pilotage d'équipements (lumières, stores) que la sécurité (surveillance, détection d'incendie) ou même l'intelligence ambiante (adaptation de l'atmosphère d'une pièce en fonction de l'activité, etc.). Déployer dans un logement toutes ces fonctions d'un seul coup peut s'avérer extrêmement onéreux. Il serait beaucoup plus amusant de concevoir une architecture modulaire permettant d'intégrer petit à petit des composants du commerce, et Robotics Developer Studio s'y prête naturellement. Le découpage du code sous forme de services permet d'intégrer au fur et à mesure les différentes fonctionnalités, selon le matériel disponible, de les reconfigurer dynamiquement, et on pourrait pousser le concept jusqu'à rendre le système « self-aware ». Malheureusement, construire tout ce système dans un article comme celui-là est impossible. Nous vous proposons donc de voir les principes à appliquer pour construire une telle architecture sur un exemple simple et concret : la robotisation d'une webcam (qui pourrait par ailleurs parfaitement s'intégrer dans notre système domotique !). Nous allons commencer par utiliser un service générique existant (pour récupérer l'image de la webcam), puis nous allons développer notre propre service pour gérer les fonctionnalités de mouvement de la caméra (Pan/Tilt). Ensuite nous rajouterons de la détection de mouvement et nous coordonnerons le tout depuis une petite interface graphique en WPF qui nous permettra de suivre ce que fait notre système. Nous aurons ainsi couvert les différentes étapes : utilisation de l'existant, développement d'un nouveau service, intégration dans une application « globale » et ajout de fonctionnalités supplémentaires.

## UTILISER DES SERVICES EXISTANTS POUR CONNECTER UN ÉQUIPEMENT STANDARD

Robotics Developer Studio dispose d'un certain nombre de services génériques, sur lesquels on peut brancher du matériel existant. Le service Webcam en est le meilleur exemple : il supporte différents formats d'image et de codecs, et permet, d'utiliser n'importe quelle webcam du commerce pour récupérer une image et l'intégrer dans une architecture Robotics Studio. Ce service a même été dérivé pour créer un service IP Camera qui nous permettrait d'utiliser une caméra IP robotisée plutôt qu'une caméra USB. Dans notre cas, nous allons utiliser une caméra Logitech Quickcam Sphere AF, celle-ci ayant déjà été étudiée par de nombreux développeurs, on trouve facilement des wrappers .NET pour la piloter, ce qui simplifiera notre intégration des fonctions Pan & Tilt par la suite.

Le service webcam démarre et trouve normalement tout seul la

caméra : pas besoin d'une intégration compliquée, donc, mais il est bon de se pencher sur les différents fichiers qui servent à configurer ce service. Le premier est le **manifest** : il sert à déclarer quel service démarre, et éventuellement la configuration qu'on lui associe. Vous trouverez le contenu de ce manifest sur notre site web. Il mentionne un fichier de configuration initial, qui permettra de spécifier les paramètres avec lequel démarrer le service : taille et qualité de l'image, format de compression, etc. : voici un extrait de ce fichier :

```
<?xml version="1.0" encoding="utf-8"?>
<WebCamState xmlns:d="http://schemas.microsoft.com/xw/2004/10/dssp.html" xmlns:s="http://www.w3.org/2003/05/soap-envelope" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns="http://schemas.microsoft.com/robotics/2006/05/multidevicewebcamservice.html">
  [...]
  <ImageSize xmlns="http://schemas.microsoft.com/robotics/2006/05/webcamservice.html">
    <X xmlns="http://schemas.microsoft.com/robotics/2006/07/physicalmodel.html">640</X>
    <Y xmlns="http://schemas.microsoft.com/robotics/2006/07/physicalmodel.html">480</Y>
  </ImageSize>
  <ViewAngle xmlns="http://schemas.microsoft.com/robotics/2006/05/webcamservice.html">0</ViewAngle>
  <Quality xmlns="http://schemas.microsoft.com/robotics/2006/05/webcamservice.html">0</Quality>
  <LastFrameUpdate xmlns="http://schemas.microsoft.com/robotics/2006/05/webcamservice.html">2010-04-26T13:33:49.3891102Z</LastFrameUpdate>
  <Cameras>
    <CameraInstance>
      <FriendlyName>QuickCam Orbit/Sphere AF</FriendlyName>
      <DevicePath>\\?\usb#vid_046d&amp;pid_0994&amp;mi_00#6&amp;1163b1be&amp;0&amp;0000#{65e8773d-8f56-11d0-a3b9-00a0c9223196}\{bbefb6c7-2fc4-4139-bb8b-a58bba724083}</DevicePath>
      <SupportedFormats>
        <Format>
          <Width>640</Width>
          <Height>480</Height>
          <MinFramesPerSecond>0</MinFramesPerSecond>
          <MaxFramesPerSecond>0</MaxFramesPerSecond>
          <Compression />
        </Format>
      </SupportedFormats>
    </CameraInstance>
  </Cameras>
  [...]
</WebCamState>
```

```
<Format>
  <Width>640</Width>
  <Height>480</Height>
  <MinFramesPerSecond>0</MinFramesPerSecond>
  <MaxFramesPerSecond>0</MaxFramesPerSecond>
  <Compression />
</Format>
<InUse>false</InUse>
</Selected>
<FramesOnDemand>false</FramesOnDemand>
</WebCamState>
```

Pour explorer ce manifest de façon plus intuitive qu'en lisant du XML dans Notepad, Robotics Developer Studio nous fournit le DSS Manifest Editor : Voici la vue qu'il offre pour le manifest le plus simple d'utilisation du service webcam (situé dans Samples/Config/webcam.manifest.xml) : [Fig.1]. On peut donc voir qu'il n'y a qu'un seul service déclaré dans ce manifest, et qu'il peut être configuré pour se servir de la caméra Logitech ou bien de la caméra intégrée de l'ordinateur portable. Nous allons laisser la configuration telle qu'elle est, mais nous pourrions ici modifier la taille de l'image capturée, le format, la webcam à utiliser, etc. Tel qu'il est configuré ce service marche, mais c'est à chacun de voir comment adapter sa configuration à sa machine. Il est donc très facile d'utiliser un service existant !

## AJOUTONS LES FONCTIONS PAN & TILT DE LA WEBCAM

La webcam Logitech QuickCam Sphere AF est motorisée, et permet donc 3 types de déplacement de la caméra : le *Pan* (rotation de gauche à droite sur un axe vertical), le *Tilt* (rotation de haut en bas sur un axe horizontal). Logitech fournit des API natives, accessibles depuis DirectShow, mais nous allons nous servir d'une API tierce plus simple, afin de limiter la complexité du code : il s'agit de la librairie native logimove.dll qu'on peut trouver à cette adresse : <http://comrade.ownz.com/projects/logimove.html>

Nous allons créer un service spécial qui ne s'occupera que des fonctions Pan & Tilt : d'abord parce que ces mouvements sont des classiques de la robotique qui peuvent s'adapter à bien d'autres situations (articulations de robots par exemple) mais aussi parce que nous voulons que notre service webcam reste générique et toutes les caméras n'intègrent pas cette fonctionnalité. Ainsi, en cas de changement ou d'intégration d'une webcam non motorisée on pourra réutiliser le composant webcam sans les fonctions Pan & Tilt.

### 1<sup>re</sup> étape : création du service

Il y a deux manières de créer un nouveau service avec Robotics Studio : soit en ligne de commande (depuis le DSS Command Prompt) avec l'utilitaire dssnewservice, soit avec Visual Studio directement, en utilisant un template de projet DSS Service. Nous allons utiliser cette deuxième méthode afin de regarder les différentes étapes de ce wizard spécial qui commence déjà à écrire le code du service pour nous : La première étape est un standard de Visual Studio : choisir le bon template et nommer le projet : [Fig.2].

Ensuite un wizard spécial démarre, et nous propose 3 onglets : le premier décrit les caractéristiques et les identifiants uniques de notre service : [Fig.3]. Nous allons laisser les valeurs par défaut. Chose intéressante : la possibilité d'utiliser un « subscription manager » qui va gérer les partenariats avec les autres services et l'envoi de notifications à nos services partenaires sans que nous ayons à

nous en préoccuper. Ce code sera généré automatiquement pour nous à la création du projet. Le deuxième onglet nous permet de déclarer des partenaires. En l'occurrence ici et pour l'instant nous n'en avons pas besoin : nous laissons donc les champs blancs. [Fig.4] La dernière étape consiste à choisir des contrats alternatifs. Pas besoin ici non plus de remplir quoi que ce soit. Les contrats alternatifs permettent de développer un service qui a ses propres spécificités mais qui répondrait également à des commandes génériques : par exemple tous les trains roulants de robots doivent répondre à des commandes de type « avance » ou « désactive-toi ». Si on développait notre propre train roulant, nous pourrions choisir comme « alternate contract » un contrat de type « Generic Differential Drive » et dans ce cas, les commandes « standard » auxquelles il faudrait répondre seraient automatiquement ajoutées dans notre code. Ensuite, libre à nous d'implémenter nos propres fonctionnalités supplémentaires comme le support de roues encodeuses, etc. Une fois que ces étapes ont été remplies, Visual Studio génère notre solution et le code nécessaire. Nous pouvons donc passer à l'implémentation.

### 2<sup>e</sup> étape : implémentation du service Pan/Tilt

La caméra dispose de 2 axes de rotation, ce qui nous donne 4 directions de mouvements basiques : nous allons donc implémenter 4 commandes pour notre service : *PanLeft*, *PanRight*, *TiltUp*, *TiltDown*. Nous allons en plus rajouter une commande *ResetPOV* qui remettra la caméra à sa position neutre. Les services interagissent entre eux par l'intermédiaire de ces commandes, ou opérations, qu'on leur envoie à travers des ports. Nous allons donc définir un nouveau type d'opération, *PanTilt*, dans notre service. Le corps de cette opération est constitué d'une requête, qui sera sérialisée lors de la communication entre les services et qui contiendra une des cinq commandes possibles. À ce titre, cette requête est ce qu'on appelle un DataContract, de la même manière que l'état du service est un DataContract pour l'opération Get (qui récupère l'état courant du service). Le code qui définit cette requête et cette opération s'ajoutent dans le fichier PanTiltCamControlTypes.cs est le suivant :

```
public enum PanTiltMovement
{
    PanLeft,
    PanRight,
    TiltUp,
    TiltDown,
    ResetPOV
}
```

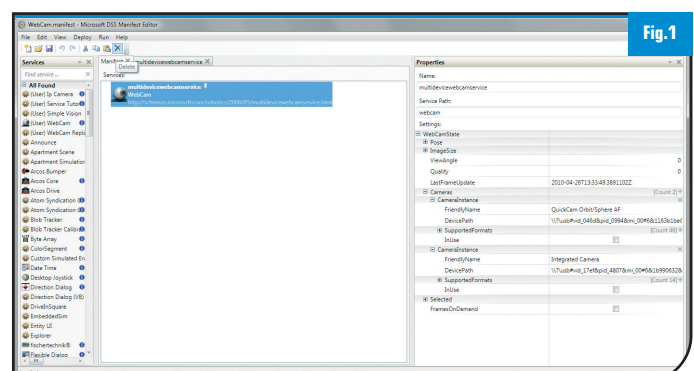


Fig.1

```
[DataContract]
public class PanTiltRequest
{
    [DataMember]
    public PanTiltMovement Move;

    public PanTiltRequest()
    { }

    public PanTiltRequest(PanTiltMovement _move)
    {
        Move = _move;
    }
}

public class PanTilt : Update<PanTiltRequest, PortSet<DefaultUpdateResponseType, Fault>>
{
    public PanTilt()
        : base(new PanTiltRequest())
    { }

    public PanTilt(PanTiltMovement _move)
        : base(new PanTiltRequest(_move))
    { }
}
```

Vous noterez que nous définissons une énumération des différentes directions que peut prendre un mouvement de type Pan/Tilt, et que c'est une instance de ce type énuméré qui constituera le corps de la requête. N'oublions pas de rajouter cette commande à la liste des opérations supportées par notre service !

```
[ServicePort]
public class PanTiltCamControlOperations : PortSet<DssspDefaultLookup, DssspDefaultDrop, Get, Subscribe, PanTilt>
{
}
```

Il faut ensuite associer un handler à cette opération qui arrivera sur le port principal de notre service. Ce handler se rajoute dans le fichier PanTiltCamControl.cs. En voici le code :

```
[ServiceHandler(ServiceHandlerBehavior.Exclusive)]
public void PanTiltHandler(PanTilt request)
{
    switch(request.Body.Move)
    {
        case PanTiltMovement.PanLeft:
            LogInfo("Requested Move: Pan Left");
            MoveLeft(_state.CameraHandle);
            break;
        case PanTiltMovement.PanRight:
            LogInfo("Requested Move: Pan Right");
            MoveRight(_state.CameraHandle);
            break;
        case PanTiltMovement.TiltDown:
            LogInfo("Requested Move: Tilt Down");
            MoveBottom(_state.CameraHandle);
            break;
        case PanTiltMovement.TiltUp:
            LogInfo("Requested Move: Tilt Up");
            MoveTop(_state.CameraHandle);
            break;
        case PanTiltMovement.ResetPOV:
            LogInfo("Requested Move: Reset POV");
            MoveHome(_state.CameraHandle);
            break;
        default:
            LogError("I don't know this move");
            break;
    }

    request.ResponsePort.Post(DefaultUpdateResponseType.Instance);
}
```

Nous ne nous appesantirons pas sur les détails de l'interop entre la DLL native et le code managé du service : il s'agit de code P/Invoke standard dont vous pourrez retrouver les détails dans les sources fournies avec cet article. Les méthodes MoveLeft, MoveRight, MoveBottom, MoveTop et MoveHome sont définies dans la librairie logimove.dll. Nous avons un dernier point à traiter pour finaliser notre service : il faut, bien sûr, initialiser la caméra au démarrage : pour cela il convient de rajouter un peu de code dans la méthode Start(), toujours dans le fichier PanTiltCamControl.cs :

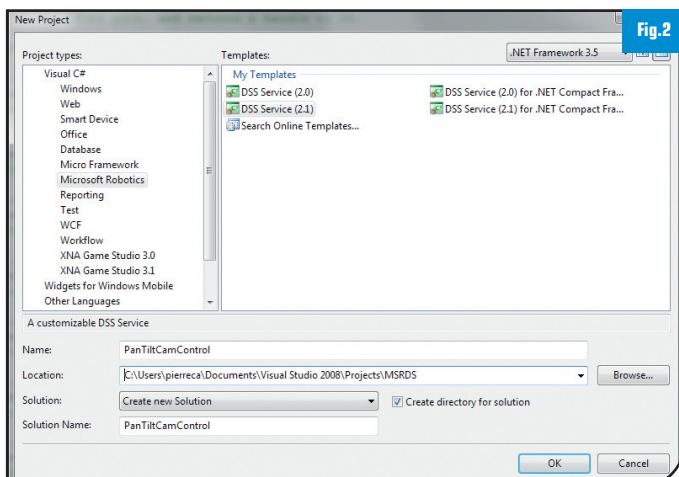


Fig.2

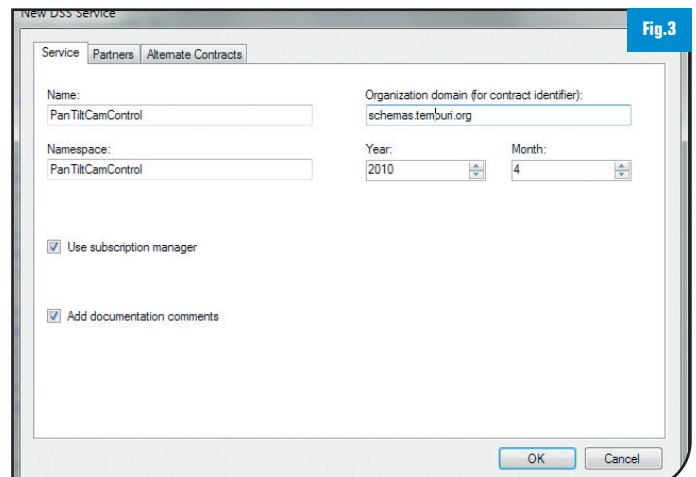


Fig.3



```
protected override void Start()
{
    int NumCams = GetDeviceCount();
    LogInfo("Nombre de caméras: " + NumCams);

    // Look for a camera
    IntPtr path = IntPtr.Zero;
    for (int i = 0; i < NumCams; i++)
    {
        path = GetDevicePath(i);
        LogInfo("Camera trouvée: " + path.ToString());
    }

    // Open the right device here
    path = GetDevicePath(1);
    _state.CameraHandle = OpenDevice(path);

    if (_state.CameraHandle != 0)
    {
        LogInfo("Utilisation de la caméra: " + _state.CameraHandle);
    }
    else
    {
        LogError("Pas de caméra, pas de chocolat!");
    }

    base.Start();
}
```

Ce code est également constitué avant tout d'appels à la librairie logi-move.dll. On peut noter l'utilisation de LogInfo et LogError pour afficher des messages d'erreur dans la console de Microsoft Robotics Developer Studio. En partant d'un service vide, nous avons donc rajouté une opération et la requête associée, ainsi que le handler pour traiter cette requête. Ce sont les étapes de base qui permettent de transformer du code standard, qu'il soit déjà managé ou encore natif, en code d'un service qui sera intégrable dans un nœud DSS et donc une infrastructure Microsoft Robotics Developer Studio.

*La suite le mois prochain...*

■ Pierre Cauchois

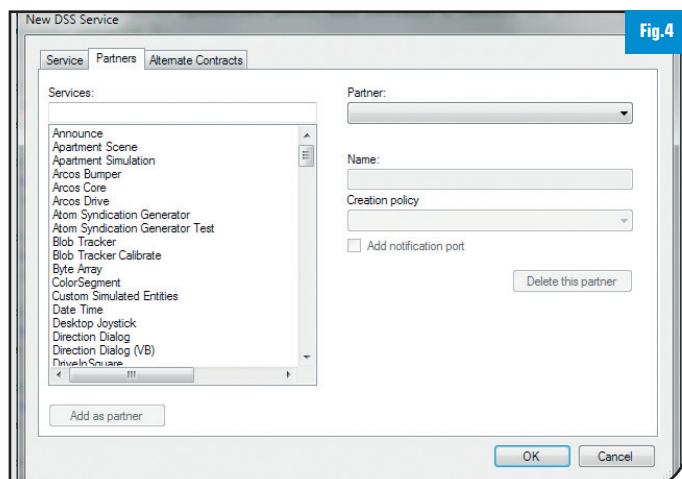


Fig.4

# Génération ROBOTS



**Et si vous passiez à la programmation d'un robot ?**

[www.generationrobots.com](http://www.generationrobots.com)

Kits robots programmables, capteurs, accessoires, supports de formation, logiciels de programmation, tutoriaux...

# Partez à la découverte de DirectX 11

Contrairement à DirectX 10, DirectX 11 ne change pas radicalement par rapport à son prédécesseur mais plutôt ajoute des fonctionnalités supplémentaires tant au niveau du rendu que de l'API elle-même. Parmi ces nouveautés, certaines sortent du lot comme les nouveaux shaders de tessellation qui permettent de jouer sur la complexité d'un objet à la volée (à nous la gestion des nurbs et autres surfaces paramétriques) !



Toutefois, l'objectif de cet article va être de proposer un tutorial simple qui servira de socle par la suite à la mise en place de fonctionnalités plus avancées.

## OUTILS DE DÉVELOPPEMENT

Dans le cadre de cet article, nous allons développer en C# en utilisant SlimDX. Ce wrapper open source constitue un excellent moyen pour faire du DirectX (de 9 à 11) efficace. Il existe également une solution fournie par Microsoft qui se nomme Windows API Code Pack. Cette dernière encapsule les nouvelles fonctionnalités de Windows 7 (dont DirectX 11) mais possède l'inconvénient de ne pas être complète pour le moment (notamment l'absence des classes de gestion des effets).

## INITIALISATION

Pour un utilisateur averti de DirectX 10, la mise en place d'un rendu DirectX 11 n'est pas fondamentalement nouvelle. Il est cependant intéressant de s'arrêter sur un service nouveau de DirectX 11 : la gestion des « features ». Sous DirectX 10, le fonctionnement était binaire : soit la carte graphique gère DirectX 10 et toutes les fonctionnalités étaient disponibles, soit la carte ne le gère pas et donc il était impossible de lancer quoi que ce soit.

Avec DirectX 11, la donne change puisque ce dernier sait fonctionner sur une carte DirectX 9, 10, 10.1 et bien sûr 11. Pour gérer cela, les fonctionnalités ont été rassemblées en plusieurs ensembles nommés « Features ». Ainsi, lors de l'initialisation de notre système, nous allons demander la mise en place de DirectX 11 avec certaines « features » activées :

```
FeatureLevel[] levels = {
    FeatureLevel.Level_11_0,
    FeatureLevel.Level_10_1,
    FeatureLevel.Level_10_0
};
```

Ces « features » vont être nécessaires lors de la création du device DirectX 11. Ce dernier constitue l'objet central autour duquel va s'articuler notre action sur DirectX 11.

Sa mise en place va également nécessiter la définition d'une *Swap Chain*. Cette classe représente l'ensemble des espaces de travail de la carte graphique, à savoir :

- Le back buffer : c'est la mémoire principale de rendu
- Le front buffer : c'est la copie de la mémoire de travail qui est pré-

sentée à l'écran. Lorsqu'une image est prête, elle est copiée depuis le back buffer vers le front buffer.

- Les buffers additionnels : Pour gagner en fluidité, il est possible de rajouter des buffers intermédiaires entre le back buffer et le front buffer. Ces derniers peuvent ainsi permettre d'absorber les à-coups. La philosophie de DirectX 11, en ce qui concerne l'instanciation des ressources, est de fournir une structure qui décrit la classe à instancier. Au niveau de notre code, cela donne donc :

```
SwapChainDescription desc = new SwapChainDescription();
desc.BufferCount = 1;
desc.Usage = Usage.BackBuffer | Usage.RenderTargetOutput;
desc.ModeDescription = new ModeDescription(0, 0, new Rational(0, 0), Format.R8G8B8A8_UNorm);
desc.SampleDescription = new SampleDescription(1, 0);
desc.OutputHandle = parentForm.Handle;
desc.IsWindowed = true;
desc.SwapEffect = SwapEffect.Discard;

Device.CreateWithSwapChain(null, DriverType.Hardware, DeviceCreationFlags.None, levels, desc, out device11, out swapChain);
```

La méthode `Device.CreateWithSwapChain` permet de générer à la fois le device et la swap chain.

## CONTEXTE IMMÉDIAT ET CONTEXTE DIFFÉRÉ

DirectX 11 supporte deux modes de fonctionnement pour l'envoi des ordres à la carte graphique. Comme ses illustres prédécesseurs, DirectX 11 comporte un mode immédiat, c'est-à-dire que les ordres sont stockés dans un buffer spécifique de la carte et envoyés d'un coup pour le rendu. Ainsi chaque image est composée d'une suite d'ordres et pour construire l'image suivante, il faut avoir envoyé la précédente (ce qui est pénalisant pour la parallélisation).

Par contre, en plus de ce mode immédiat, DirectX 11 supporte un mode différé. Dans ce mode, le développeur peut enregistrer ses ordres pour une image sans pour autant avoir envoyé l'image précédente. Il est ainsi possible d'avoir plusieurs buffers d'ordres construits en parallèle pendant qu'un autre buffer est en cours de rendu : [Fig.1]

Dans notre exemple, nous allons nous contenter du mode immédiat :

```
immediateContext = device11.ImmediateContext;
```

C'est donc via cette instance que nous communiquerons nos ordres à la carte graphique.

Une des premières informations à donner à la carte graphique est

le buffer de rendu actif. Pour ce faire nous allons demander à la Swap Chain de nous retourner son back buffer :

```
backBuffer = Resource.FromSwapChain<Texture2D>(swapChain, 0);

// Definition de la cible de rendu
renderView = new RenderTargetView(device11, backBuffer);
immediateContext.OutputMerger.SetTargets(renderView);
```

## LES RESSOURCES ET LA NOTION DE VUE

DirectX 10 a introduit la notion de ressources unifiées et de vue. Jusqu'alors les ressources (textures, vertex buffer, index buffer, render targets, etc...) étaient toutes traitées de manière différente alors que dans l'absolu il ne s'agissait finalement que d'espaces dans la mémoire de la carte graphique.

La notion de ressources unifiées permet de n'avoir qu'un seul type de ressources : Les buffers mémoires. Les vues permettent alors (sur une même ressource) de mettre en place un système de différenciation (RenderTargetView, ShaderResourceView, etc...).

Dans le cadre de DirectX 11 rien ne change de ce côté-là puisque ces notions s'appliquent de la même manière.

Ainsi le code de la création d'une ressource pour notre texture se passera ainsi :

```
Texture2D texture2D = Texture2D.FromFile(device11, "yoda.jpg");
ShaderResourceView view = new ShaderResourceView(device11,
texture2D);
```

La classe Texture2D hérite de la classe Resource et pour obtenir une vue dessus qui nous permettra de la transmettre à la classe, on passe tout simplement par une instance de la classe ShaderResourceView.

## LES SHADERS

DirectX 11 (tout comme DirectX 10) ne fonctionne que sur une base de shaders. Ces derniers sont des programmes qui s'exécutent sur la carte graphique et dont le but est de mettre en place l'image finale.

Toutefois, DirectX 11 a rajouté plusieurs nouveaux types de shaders

dont les shaders DirectCompute qui permettent d'utiliser la puissance de la carte graphique pour faire des calculs mathématiques.

Dans le cadre de cet article nous allons nous concentrer sur les deux premiers types de shaders : Les vertex shaders et les pixels shaders. Les vertex shaders ont la responsabilité de convertir les vertices (les points 3D) en pixels. Les pixels shaders doivent quant à eux donner la couleur de chaque pixel.

Une image pour être rendue complètement doit donc avoir un shader de chaque catégorie. DirectX 11 fournit pour cela plusieurs solutions. La solution la plus simple dans notre cas est de passer par le framework Effect (qui est inclus dans l'API) qui permet dans un seul fichier .fx de mettre en place tout le système de rendu.

Dans notre cas, ce fichier va tout d'abord contenir une déclaration des variables globales :

```
cbuffer globals
{
    matrix worldProjectionMatrix;
}

Texture2D diffuseTexture;
SamplerState diffuseSampler
{
    Filter = MIN_MAG_MIP_LINEAR;
    AddressU = Wrap;
    AddressV = Wrap;
};
```

On voit apparaître un bloc de variables (nommé globals) qui dans notre cas ne contient qu'une matrice (celle qui permettra de transformer et de projeter nos vertices à l'écran). On voit également une texture (diffuseTexture) et un sampler (diffuseSampler). Ce dernier se charge de fournir une instance pour configurer la lecture dans une texture (notre sampler est notamment configuré ici pour activer le filtrage tri-linéaire).

Après avoir défini les variables, le fichier permet de décrire la structure des vertices qui seront passés au vertex shader ainsi que la structure des pixels qui seront transmis par le vertex shader au pixel shader :

```
struct VS_IN
{
    float3 pos : POSITION;
    float2 uv : TEXCOORD0;
};

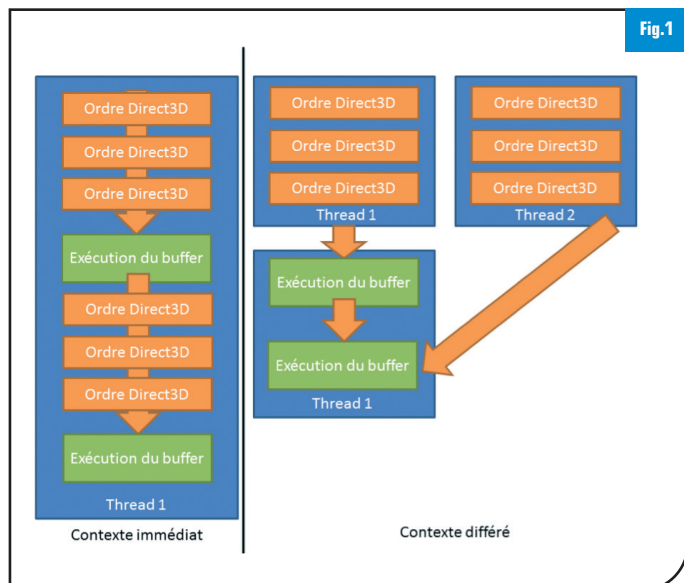
struct PS_IN
{
    float4 pos : SV_POSITION;
    float2 uv : TEXCOORD0;
};
```

Dans notre exemple, nous nous contenterons de la position et d'une coordonnée de texture.

Finalement, le fichier contient le vertex shader, le pixel shader ainsi que la définition de la technique :

```
// Vertex Shader
```

Fig.1





```

PS_IN VS( VS_IN input )
{
    PS_IN output = (PS_IN)0;

    output.pos = mul(float4(input.pos, 1), worldProjectionMatrix);
    output.uv = input.uv;

    return output;
}

// Pixel Shader
float4 PS( PS_IN input ) : SV_Target
{
    return diffuseTexture.Sample(diffuseSampler, input.uv);
}

// Technique
technique10 Render
{
    pass P0
    {
        SetGeometryShader( 0 );
        SetVertexShader( CompileShader( vs_4_0, VS() ) );
        SetPixelShader( CompileShader( ps_4_0, PS() ) );
    }
}

```

La technique permet de spécifier le compilateur à utiliser (ici 4.0 donc pour une carte DirectX 10.0 minimum).

Ce fichier .fx va par la suite être lu par notre code, transmis à DirectX ou le framework Effect le compilera pour l'envoyer à la carte graphique :

```

// Compilation du shader
using (ShaderBytecode byteCode = ShaderBytecode.
CompileFromFile("Simple.fx", "bidon", "fx_5_0",
                ShaderFlags.
OptimizationLevel3, EffectFlags.None))
{
    effect = new Effect(device11, byteCode, Effect
Flags.None);
}
const int vertexSize = 20;
var technique = effect.GetTechniqueByIndex(0);
pass = technique.GetPassByIndex(0);

```

On voit dans le code ci-dessus que via la classe ShaderByteCode nous allons compiler notre fichier .fx en byte code (code machine de la carte graphique). Ce dernier sera alors utilisé pour créer une instance de la classe Effect qui va nous permettre de récupérer notre technique puis l'unique passe (la conjonction de notre vertex shader et pixel shader).

## DÉFINITION DE L'OBJET 3D

Tout bon moteur 3D n'est rien s'il n'a rien à afficher. Pour notre exemple nous allons travailler avec un simple plan que nous allons créer. Toujours dans le cadre des ressources unifiées, notre plan va être constitué de deux buffers mémoires : un pour les 4 points (ver-

tex buffer) et un pour définir les deux faces (index buffer) : [Fig.2]

Ce qui en code se traduit par :

```

// Generation du plan
var stream = new DataStream(4 * vertexSize, true, true);
stream.WriteRange(new[]
{
    -1.0f, -1.0f, 0f, 0f, 1.0f,
    1.0f, -1.0f, 0f, 1.0f, 1.0f,
    1.0f, 1.0f, 0f, 1.0f, 0.0f,
    -1.0f, 1.0f, 0f, 0.0f, 0.0f,
});
stream.Position = 0;

var vertices = new SlimDX.Direct3D11.Buffer(device11,
stream, new BufferDescription
{
    BindFlags = BindFlags.VertexBuffer,
    CpuAccessFlags = CpuAccessFlags.None,
    OptionFlags = ResourceOptionFlags.None,
    SizeInBytes = (int)stream.Length,
    Usage = ResourceUsage.Default
});
stream.Dispose();
stream = new DataStream(6 * 2, true, true);
stream.WriteRange(new[]
{
    (short)0, (short)1, (short)2,
    (short)0, (short)2, (short)3
});
stream.Position = 0;

var indices = new SlimDX.Direct3D11.Buffer(device11,
stream, new BufferDescription
{
    BindFlags = BindFlags.IndexBuffer,
    CpuAccessFlags = CpuAccessFlags.None,
    OptionFlags = ResourceOptionFlags.None,
    SizeInBytes = (int)stream.Length,
    Usage = ResourceUsage.Default
});
stream.Dispose();

```

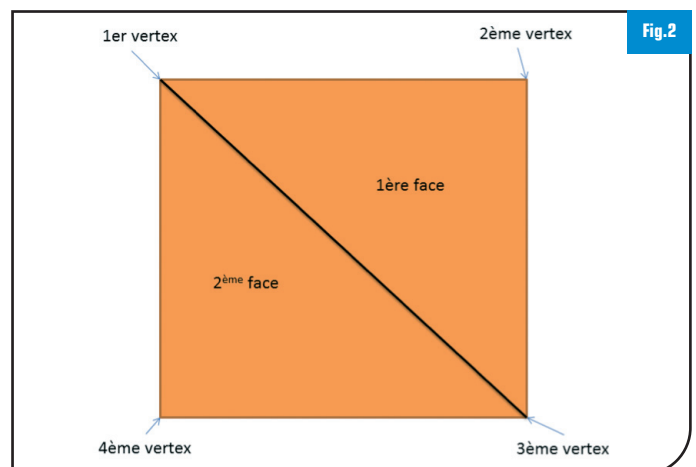


Fig.2

Une fois les deux buffers construits, il ne nous reste qu'à les indiquer comme source à DirectX 11 :

```
immediateContext.InputAssembler.PrimitiveTopology = PrimitiveTopology.TriangleList;

immediateContext.InputAssembler.SetVertexBuffers(0, new VertexBufferBinding(vertices, vertexSize, 0));
immediateContext.InputAssembler.SetIndexBuffer(indices, Format.R16_UInt, 0);
```

## LA GESTION DES ÉTATS

DirectX 11, malgré le fait qu'une grande partie de sa configuration se trouve dans les shaders, reste une machine à état. De ce fait, pour configurer son comportement il existe des structures qui ciblent telle ou telle partie du système global.

Par exemple, par défaut, le système élimine les faces cachées (les faces qui ne pointent pas vers l'utilisateur). Toutefois dans le cadre de notre exemple, nous souhaitons que le plan apparaisse des deux côtés. Il va donc falloir configurer un état puisque ce genre de comportement ne dépend pas des shaders.

La différence avec DirectX 9 se situe au niveau du rassemblement des états. Dans DirectX 9, chaque fonction avait un booléen associé et il était donc possible de changer les états unitairement. Toutefois

ce fonctionnement entraînait des pertes de performances et le développeur devait être attentif à ne pas trop souvent changer les états. Depuis DirectX 10, les états sont rassemblés par groupes fonctionnels et sont stockés dans des structures en mémoire, ce qui minimise fortement l'impact d'un changement d'état.

Ainsi dans notre code si l'on souhaite changer un état, cela se résume à instancier une structure associée et affecter cette structure à DirectX 11 :

```
RasterizerStateDescription rasterizerStateDescription
= new RasterizerStateDescription();

rasterizerStateDescription.CullMode = CullMode.None;

immediateContext.Rasterizer.State = RasterizerState.FromDescription(device11, rasterizerStateDescription);
```

## PHASE DE RENDU

Contrairement à DirectX 9, la phase de rendu est extrêmement simple et se résume à activer la passe et à donner l'ordre de rendu :

```
pass.Apply(device11.ImmediateContext);
device11.ImmediateContext.DrawIndexed(6, 0, 0);
swapChain.Present(0, PresentFlags.None);
```

Il est à noter que les ordres sont envoyés effectivement à la carte graphique lors du *Present*. Ce dernier effectue également le *swap* c'est-à-dire la présentation du buffer de travail et la mise en place d'un nouveau buffer de travail.

## MISE EN PLACE DU DEBUG

Pour finir, il est un point qu'il est important de maîtriser pour bien réussir un développement avec DirectX 11 : le debug.

Pour activer le debug et plus précisément les informations relatives aux erreurs fournies par DirectX 11 lors de problèmes, il faut lancer le DirectX Control Panel qui se trouve dans les outils installés par le SDK. Ce dernier permet de contrôler l'activation de la couche de debug et permet de récupérer des messages de DirectX 11 dans notre console de sortie au niveau de Visual Studio 2008 : [Fig.3].

Attention toutefois à autoriser le débogage d'applications non gérées au sein des propriétés du projet Visual Studio.

## LIENS UTILES

Homepage DirectX pour les développeurs :

<http://msdn.microsoft.com/en-us/directx/default.aspx>

SlimDX : <http://slimdx.org/>

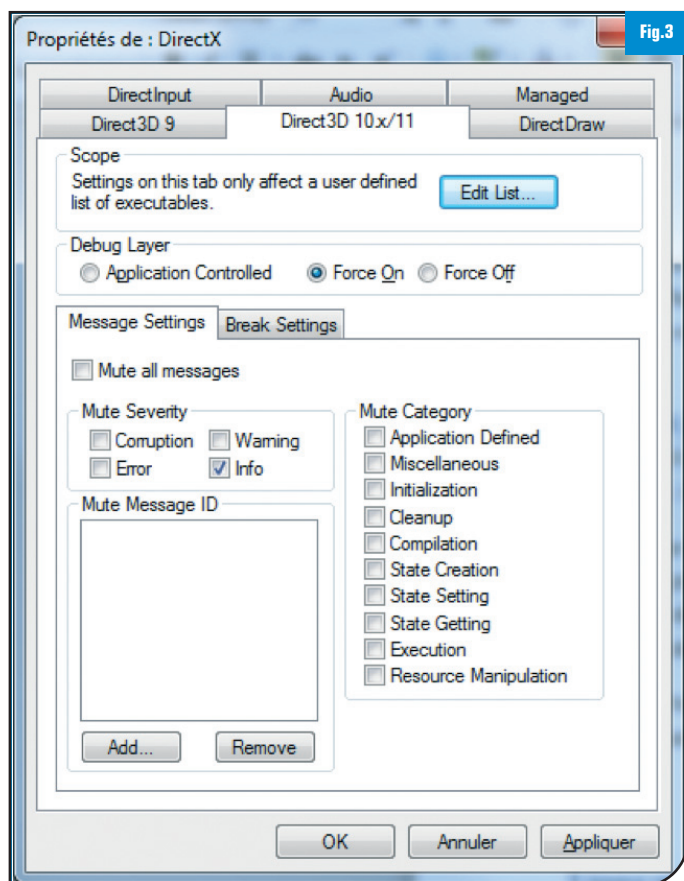
Forums XNA pour DirectX 11 :

<http://forums.xna.com/forums/76.aspx>

Windows API Code Pack :

<http://code.msdn.microsoft.com/WindowsAPICodePack>

■ David Catuhe



# L'information permanente

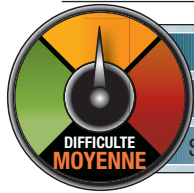
● L'actu de Programmez.com : le fil d'info quotidien

● La newsletter hebdo : la synthèse des informations indispensables. Abonnez-vous, c'est gratuit !

[www.programmez.com](http://www.programmez.com)

# Interroger Google Analytics

Google Analytics est un service bien connu des webmasters, leur permettant de connaître la fréquentation de leur site depuis un navigateur. Mais Google Analytics permet aussi d'accéder à ces données par programmation, ce que nous faisons aujourd'hui.



APPLICATION : WEB

LANGAGE : PYTHON

SOURCE : OUI

Consulter les statistiques de fréquentation de son site fournies par Google Analytics depuis un navigateur, c'est bien. Mais il est des situations où obtenir des données par programmation c'est mieux. Google Analytics expose une API qui permet cela. Il s'agit d'une API en REST, ainsi que pour nombre des services Google. Google propose des bibliothèques pour travailler avec cette API. Ces bibliothèques, qui existent en Java, Javascript et Python rendent transparent le travail avec une architecture REST, ce qui est à la fois leur avantage et leur inconvénient. C'est un inconvénient si l'on veut comprendre le fonctionnement des choses et communiquer avec les API depuis un autre langage. En effet REST n'étant basé que sur le protocole HTTP, tout langage qui sait ouvrir un socket sécurisé est un langage qui va bien. Ainsi, au cours de cet article, si nous utilisons Python et Java, nous n'utiliserons pas de bibliothèque, mais nous travaillerons fondamentalement en REST, ce qui permettra au lecteur intéressé de transposer facilement vers "son" langage :). Le lecteur trouvera les sources de nos exemples sur le site [www.programmez.com](http://www.programmez.com).

## 1 QU'EST-CE QUE REST ?

REST, ou REpresentational State Transfert n'est pas un langage, même si précédemment on a parlé, peut-être par abus de langage, d'API en REST. REST est une architecture, qui n'est rien d'autre que l'architecture du Web. En dépit des apparences, sur le Web, tout n'est pas page, mais tout est ressource. Une ressource peut être rendue de différentes manières, par exemple avec de l'HTML, et on aura alors une page Web classique, visualisée dans le navigateur, ou bien en XML, ou de beaucoup d'autres façons. C'est le rendu d'une ressource qui voyage sur le réseau, et non la ressource, d'où le nom de Representational State Transfert. Une ressource est identifiée par une URI ou identifiant unique. On y accède par une URL, qui permet de localiser la ressource, et le protocole de communication. Ainsi dans <http://www.programmez.com/index.php>

La ressource est index.php, un fichier qui contient du code PHP. La ressource est localisée dans le domaine [www.programmez.com](http://www.programmez.com) et le protocole est http. Cette URL tapée dans le navigateur, celui-ci demande, via le protocole http, le rendu de index.php, ce qui aboutit à la page d'accueil de votre site préféré dans votre navigateur, et non au code source PHP :)

## 2 LE PROTOCOLE HTTP

HTTP est un protocole de communication client-serveur. Sa version 1.0, nécessaire et suffisante pour notre propos est définie dans un document portant le doux nom de RFC 1945. Ce document est accessible à <http://www.ietf.org/rfc.html>. A notre niveau, nul besoin

d'étudier un tel document dans le détail. Voici ce que nous devons savoir: Dans ce protocole, le client (usuellement le navigateur) envoie une chaîne de caractères vers le serveur à travers un socket, sécurisé pour nous avec Google Analytics, pour d'évidentes raisons de confidentialité. Cette chaîne, qui doit être constituée selon des règles précises s'appelle une requête. Le serveur analyse alors la requête et y répond, également en envoyant dans le socket une réponse, également constituée selon une règle précise. Schématiquement une requête http est constituée comme ceci:

```
Ligne de commande (Commande, URL, Version de protocole)
En-tête de requête
[Ligne vide]
Corps de requête
```

Il existe de nombreuses commandes. Nous n'avons aujourd'hui besoin que des commandes GET et POST. Bien remarquer la présence de la ligne vide. L'omettre par erreur fait échouer la communication et s'arracher les cheveux :-)

## 3 DANS LE CAS DE GOOGLE ANALYTICS

Pour obtenir des données du service Google Analytics, il faut procéder en deux étapes. Une étape d'authentification, puis l'étape où l'on interroge le service. L'étape d'authentification se fait au moyen d'une commande POST, puis les interrogations au moyen d'une commande GET. Lorsque l'identification réussit, le service fournit un token qui devra être utilisé pour les interrogations. Ce token est une chaîne de caractères vertigineuse. Sa validité est de 14 jours. Bien entendu une documentation complète est fournie par Google à cette adresse : <http://code.google.com/intl/fr-FR/apis/analytics/>

## 4 AUTHENTIFICATION AUX SERVICES GOOGLE

Google propose en fait trois méthodes différentes d'authentification. Nous utilisons la méthode ClientLogin. La documentation nous apprend que cette ressource se situe à /accounts/ClientLogin, sous le domaine [www.google.com](http://www.google.com). Nous devons donc envoyer une requête POST à <http://www.google.com/accounts/ClientLogin>. Très logiquement la requête doit transmettre deux paramètres au serveur : une adresse email et un mot de passe. En outre nous devons transmettre le nom du service auprès duquel nous nous identifions, ainsi qu'une chaîne arbitraire. Voici un script Python qui fait ce travail :

```
import urllib, urllib

params = urllib.urlencode({'accountType': 'HOSTED_OR_GOOGLE',
                           'Email': 'mon.mail@gmail.com',
                           'Passwd': 'sesame',
                           'service': 'analytics',
```



```
'source' : "fred-maz-1.0"}}

headers = {"Content-type": "application/x-www-form-urlencoded"}

conn = httplib.HTTPSConnection("www.google.com", 443)
conn.request("POST", "/accounts/ClientLogin", params, headers)
response = conn.getresponse()
print response.status, response.reason

data = response.read()
print data
conn.close()
```

Dans ce code, nous utilisons une classe de la librairie Python: HTTPS-Connection, qui permet de constituer la requête dont nous avons parlé plus haut, puis lire la réponse émise par le serveur. Comme illustré [Fig.1], celui-ci nous retourne des token d'identifications. Il suffit d'extraire la dernière ligne de la réponse pour pouvoir interroger Google Analytics. Outre la constitution de la requête, Python a encodé correctement nos paramètres. Voyons maintenant comment il faudrait faire sans aucune fonction ou classe pour nous aider.

## 5 CONSTITUTION D'UNE REQUÊTE HTTP

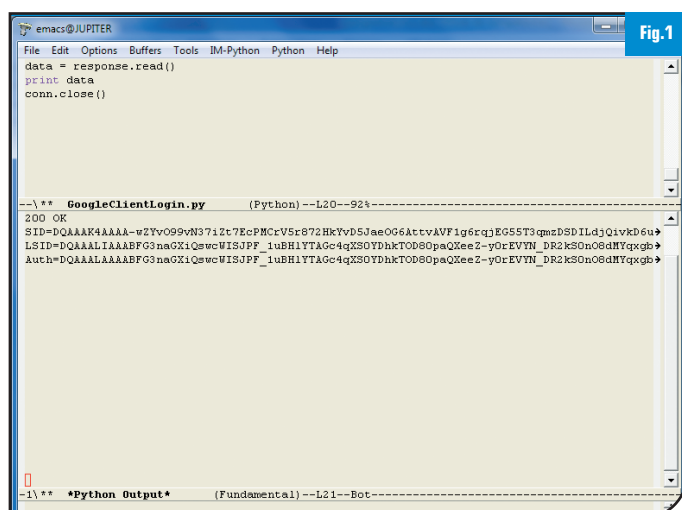
Nous utilisons maintenant Java. Nous pourrions prendre n'importe quel langage, car nous travaillons au plus bas niveau :

```
package demosocket;

import java.io.*;
import java.net.*;
import javax.net.ssl.SSLSocketFactory;

public class GoogleClientLogin {

    public static void main(String[] args) {
        String command = "POST /accounts/ClientLogin HTTP/1.0\r\n";
        String host = "Host: www.google.com\r\n";
        String action = "Connection: Close\r\n";
        String content_type =
```



Exécution de notre exemple client-login.py dans Emacs

```
"Content-type: application/x-www-form-urlencoded\r\n";

String params =
    "accountType=GOOGLE&Email=mon.mail%40gmail.com&3Passwd=sesame
    &service=analytics&source=fred-maz-1.0";

String content_length =
    Content-Length: " + params.length() + "\r\n\r\n";

String requete = command;
requete += host;
requete += action;
requete += content_type;
requete += content_length;
requete += params;
requete += "\r\n";

System.out.println(requete);

try {

    InetAddress google =
        InetAddress.getByName("www.google.com");
    Socket sub_s = new Socket(google, 443);

    SSLSocketFactory socketFactory =
        (SSLSocketFactory) SSLSocketFactory.getDefault();

    Socket s =
        socketFactory.createSocket(sub_s,
            "www.google.com", 443, true);

    BufferedReader in = new BufferedReader(
        new InputStreamReader(s.getInputStream()));

    DataOutputStream out =
        new DataOutputStream(s.getOutputStream());

    out.writeBytes(requete);

    String responseLine;
    while ((responseLine = in.readLine()) != null) {
        System.out.println(responseLine);
    }

    in.close();
    out.close();
    s.close();

} catch (UnknownHostException uex) {
    uex.printStackTrace();
} catch (IOException e) {
    System.out.println("Erreur" + e);
} catch (Exception ex) {
    ex.printStackTrace();
}

}
```

Voici la requête constituée par ce code, avant écriture dans le socket

```
POST /accounts/ClientLogin HTTP/1.0
Host: www.google.com
Connection: Close
Content-type: application/x-www-form-urlencoded
Content-Length: 102

accountType=GOOGLE&Email=mon.mail%40gmail.com&Passwd=sesame&
service=analytics&source=fred-maz-1.0
```

On y voit que la forme donnée au paragraphe 2 est strictement respectée. On y voit aussi comment les paramètres passés au serveur et placés dans le corps de la requête doivent être écrits et encodés. Comme finalement dans cet exemple le seul caractère devant être encodé est @, nous l'avons fait à la main (%40). Sinon on aurait utilisé la classe Java `URLEncoder`

## 6 ORGANISATION DE L'API GOOGLE ANALYTICS

Hormis les fonctionnalités d'authentification dont nous venons de parler, Google Analytics est organisé pour fournir deux flux: *Account Feed* et *Data Feed*, compte et données. Interroger le premier flux fournit des éléments essentiels pour interroger le second. L'interrogation de Account Feed fournit un résultat au format XML contenant essentiellement :

- des identifiants de segments prédéfinis (balises XML dxp:segment). Un segment fournit des données filtrées. Ainsi par défaut et par exemple un segment donne les nouveaux visiteurs d'un site, tandis qu'un autre donne le nombre de visites sans rebond.
- les profils associés au compte. Vous les trouverez entre les balises XML entry
- des identifiants de table. Chacun des profils contient un identifiant de table (balise XML dxp:tableId) qui permettra l'accès aux données fournies selon la configuration du profil.

## 7 INTERROGER L'ACCOUNT FEED

Voici un script Python qui interroge l'Account Feed de Google Analytics. La ressource se trouve à `feeds/accounts`. La capture [\[Fig.1\]](#) montre le script en action dans Emacs.

```
#DemoAccountFeed.py
import urllib, urllib

params = urllib.urlencode({'prettyprint': 'true'})
print params

headers = {"Content-type" : "application/x-www-form-urlencoded",
           "GData-Version" : "2",
           "Authorization" : "GoogleLogin Auth=DQAAALEAAA...jlsUHqQig"}
print headers

conn = urllib.HTTPSConnection("www.google.com", 443)
conn.request("GET", "/analytics/feeds/accounts/default?prettyprint=
true", None, headers)
response = conn.getresponse()
print response.status, response.reason

data = response.read()
print data
conn.close()
```

Voici les points importants: le token d'authentification est passé au service dans l'en-tête de la requête. Comme nous l'avons dit plus haut, la commande http POST est désormais délaissée au profit de la commande GET. Dans le cas d'une commande GET les paramètres ne sont pas placés dans le corps de la requête, mais sont directement concaténés à l'URL, après encodage, bien entendu. Le lecteur trouvera en outre sur notre site un exemple de code java qui construit une requête GET intégralement (classe AccountFeed).

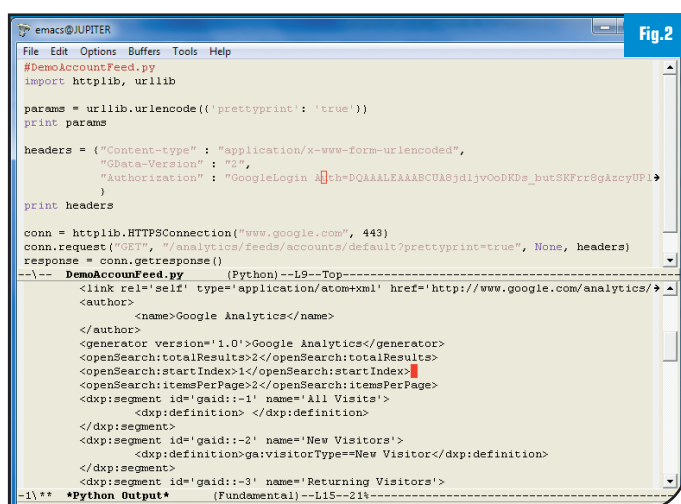
Voici dans ce cas la requête qui est générée et envoyée à Google Analytics :

```
GET /analytics/reeds/accounts/default?prettyprint=true HTTP/1.0
Host: www.google.com
Connection: Close
GData-Version: 2
Authorization: GoogleLogin auth=DQAAALEEAAA...lj1SUHqQig
Content-type: application/x-www-form-urlencoded
```

**Attention !** Cela ne se voit pas ci-dessus en raison de l'absence du corps de requête, mais il y a bien une ligne vide en fin de texte. L'absence de cette ligne ferait échouer l'opération.

## 8 INTERROGER LE DATA FEED

Nous arrivons à l'interrogation du Data Feed, c'est-à-dire à l'obtention des données fournies par Google Analytics. Au niveau de la programmation HTTP proprement dite, il n'y a rien à ajouter. Tout ce qui a été dit au paragraphe 7 à propos de requêtes http basées sur les commandes GET reste valable. Là encore, le résultat sera fourni dans un document XML. Pour qu'une requête soit valide aux yeux du service Analytics, elle doit comporter 4 paramètres au strict minimum. L'ordre dans lequel figurent ces paramètres n'a aucune importance. Ces paramètres sont : l'identifiant de table tel que mentionné précédemment, une date de début et une date de fin de la période pour laquelle on interroge le service et une métrique. Analytics travaille avec des métriques et des dimensions pour vous fournir vos données. Qu'est-ce donc ? Une métrique sert à mesurer. Par



*Interrogation de l'Account Feed de Google Analytics avec un script Python, sous Emacs.*

exemple la métrique `ga:visits` mesure le nombre de visites sur la période de temps définie dans la requête. Si par exemple on demande le nombre de visites pour une journée avec cette seule métrique, le document contiendra une seule entrée:

```
<entry gd:etag='W/&quot;Ak8EQX47...RGU0.&quot;' gd:kind='analytics
#datarow'>
  <id>http://www.google.com/analytics/feeds/data?ids=ga:xx
xxxxxx&start-date=2010-01-08&end-date=2010-01-08</id>
  <updated>2010-01-07T16:00:00.001-08:00</updated>
  <title>
  <link rel='alternate' type='text/html' href='http://www.google
.com/analytics/'>
  <dxp:metric confidenceInterval='0.0' name='ga:visits' type='
integer' value='123456'>
</entry>
```

Et la métrique nous apprend que nous avons eu 123 456 visites pour la journée du 8 janvier 2010 :) Les dimensions servent à segmenter les données mesurées pour une métrique. Par exemple la dimension `ga:browser` permet de segmenter, ou classer si l'on préfère, le nombre de visites par les navigateurs. Ainsi une requête avec `ga:visits` comme métrique et `ga:browser` comme dimension peut retourner (nous ne donnons qu'une sortie simplifiée) :

```
<dxp:aggregates>
  <dxp:metric confidenceInterval='0.0' name='ga:visits' type='
integer' value='123456'>
</dxp:aggregates>
```

Puis viendront autant d'entrées que de navigateurs différents ayant visité le site pendant la période, chaque entrée contenant le nombre de visites pour un navigateur donné, le total valant bien entendu le nombre donné dans la section `dxp:aggregates`. Il est ainsi possible d'établir des rapports très précis sur la fréquentation d'un site, mais en étant attentif au fait que toutes les combinaisons métriques/dimensions ne sont pas valides. Métriques, dimensions et combinaisons sont bien entendu entièrement documentées sur le site d'Analytics. En plus de tout cela, il est possible de filtrer les résultats et aussi de les trier. Nous renvoyons le lecteur vers la documentation d'Analytics pour ces détails. Voici pour illustrer notre propos, un script Python qui demande les visites d'un site et les segmente par navigateur (Fig.3).

```
#DemoDataFeed.py
import urllib

url_base="https://www.google.com/analytics/feeds/data"

query_params_base="?ids=ga:xxxxxx&start-date=2010-01-08&end-date=
=2010-01-08"

# Obtenir visites et visiteurs uniques
# query_params="&metrics=ga:visits,ga:visitors"

# Obtenir visites et navigateurs (visiteurs ici est une combi
naison invalide)
query_params="&metrics=ga:visits&dimensions=ga:browser&pretty
```

```
print=True"

url= url_base + query_params_base + query_params

print url

headers = {"Content-type" : "application/x-www-form-urlencoded",
           "GData-Version" : "2",
           "Authorization" : "GoogleLogin auth=DQAAALEAAA...
lSUHQig"
           }

print headers

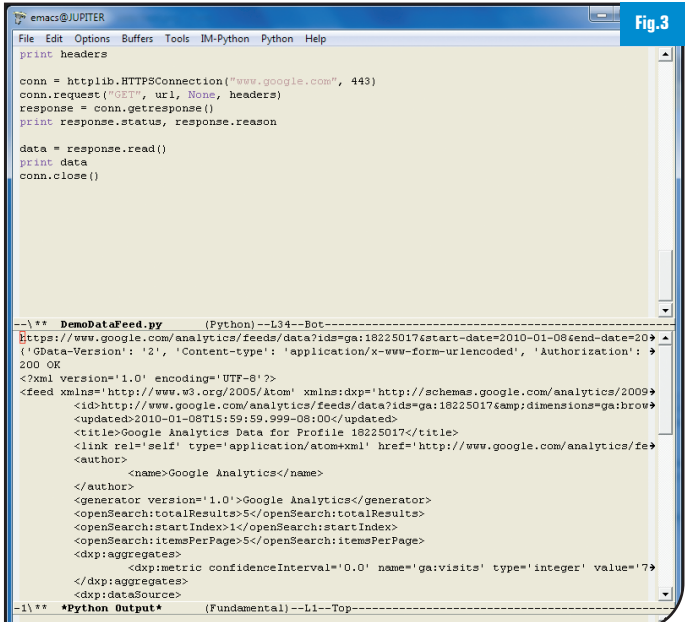
conn = urllib.HTTPSConnection("www.google.com", 443)
conn.request("GET", url, None, headers)
response = conn.getresponse()
print response.status, response.reason

data = response.read()
print data
conn.close()
```

## 9 CONCLUSION

Nous savons maintenant interroger Analytics avec n'importe quel langage. Bien sûr il est nécessaire de traiter les résultats fournis en XML. Cela sort du cadre de cet article et ne présente pas de difficultés particulières. En revanche, si l'on apprécie peu les chiffres, il est bon de savoir qu'il est possible de demander à Google de générer des diagrammes avec son API Google Chart. Celle-ci est en REST elle aussi, et les principes vus dans cet article se transposent directement :-)

■ Frédéric Mazué - [fmazue@programmez.com](mailto:fmazue@programmez.com)



```
File Edit Options Buffers Tools IM-Python Python Help
print headers

conn = urllib.HTTPSConnection("www.google.com", 443)
conn.request("GET", url, None, headers)
response = conn.getresponse()
print response.status, response.reason

data = response.read()
print data
conn.close()

--** DemoDataFeed.py (Python) --L34--Bot--
https://www.google.com/analytics/feeds/data?ids=ga:18225017&start-date=2010-01-08&end-date=2010-01-08
('GData-Version': '2', 'Content-type': 'application/x-www-form-urlencoded', 'Authorization': 'GoogleLogin auth=DQAAALEAAA...lSUHQig')
200 OK
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom' xmlns:dxp='http://schemas.google.com/analytics/2009'>
  <id>http://www.google.com/analytics/feeds/data?ids=ga:18225017&start-date=2010-01-08&end-date=2010-01-08</id>
  <updated>2010-01-08T15:59:59.999-08:00</updated>
  <title>Google Analytics Data for Profile 18225017</title>
  <link rel='self' type='application/atom+xml' href='http://www.google.com/analytics/feeds/data?ids=ga:18225017&start-date=2010-01-08&end-date=2010-01-08'>
  <author>
    <name>Google Analytics</name>
  </author>
  <generator version='1.0'>Google Analytics</generator>
  <openSearch:totalResults>5</openSearch:totalResults>
  <openSearch:startIndex>1</openSearch:startIndex>
  <openSearch:itemsPerPage>5</openSearch:itemsPerPage>
  <dxp:aggregates>
    <dxp:metric confidenceInterval='0.0' name='ga:visits' type='integer' value='7'>
  </dxp:aggregates>
  <dxp:dataSource>
  </dxp:dataSource>
</feed>

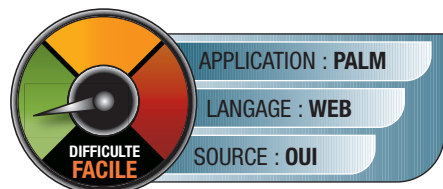
--** Python Output ** (Fundamental) --L1--Top--
```

Exécution du Data Feed de Google Analytics avec un script Python sous Emacs.



# Ecrire des applications pour Palm webOS

Récemment, Palm a rendu public Mojo, son SDK de développement d'applications pour les smartphones bâti sur son système d'exploitation Palm webOS. Découvrons les bases de son maniement. Depuis, Palm a été racheté par HP...



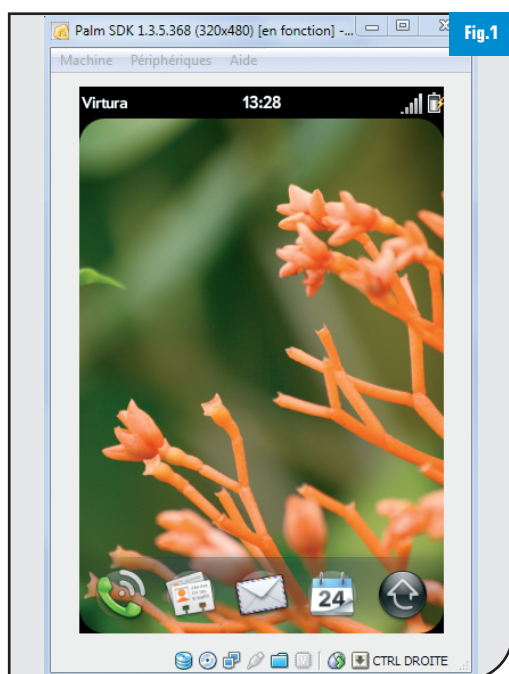
Les smartphones ont le vent en poupe en ce moment, et, naturellement, de nombreuses applications sont écrites pour eux par les professionnels

et aussi par les passionnés, lorsqu'ils peuvent avoir accès aux outils de développement. Ayant compris cela avec un peu de retard sur la concurrence, Palm a mis un SDK à disposition du public il y a quelques mois. Ce SDK, disponible pour Windows, Linux et Mac, est de bonne qualité, et, fait qui mérite d'être souligné, sa documentation aussi. Des outils supplémentaires viennent enrichir le tout. Voyons comment travailler avec.

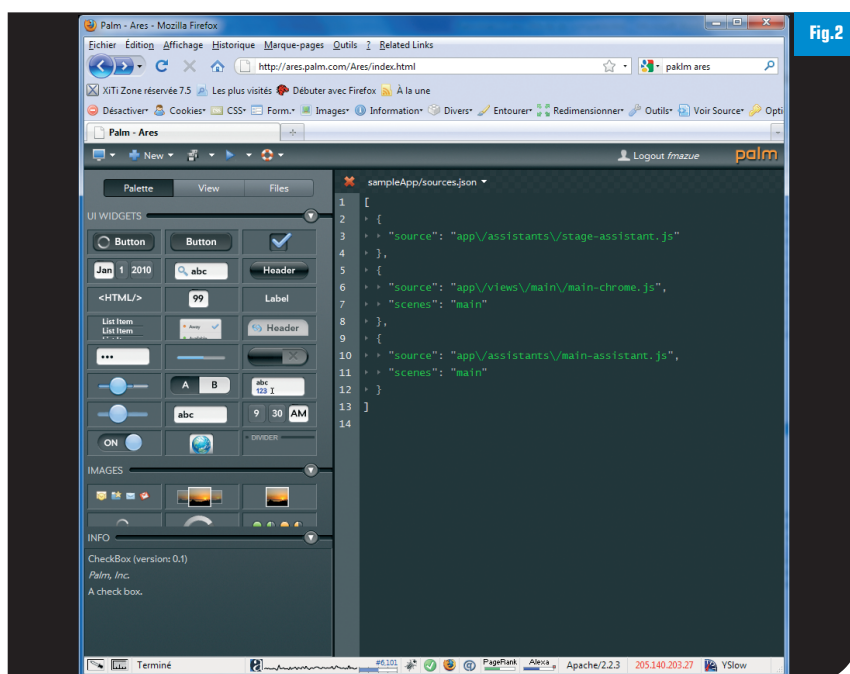
## 1 PRÉ REQUIS OUTILS ET INSTALLATIONS

Pour écrire des applications pour Palm webOS seules sont nécessaires des notions de XHTML, de CSS et de Javascript. Si vous ne maîtrisez pas encore Javascript, connaître un langage comme Java vous aidera. (Même si en dépit de leurs noms, Java et Javascript sont sur le fond très différents). Javascript n'est toutefois pas un langage difficile à aborder. Enfin, il faut connaître le format de données JSON, ce qui ne pose aucune difficulté. Sachant que Windows XP 64 bits n'est pas supporté par le SDK et que Linux 64 bits l'est avec difficulté, vous devez vous rendre sur le site de Palm, pour télécharger cet outil. Son installation requiert la présence d'une JDK

suffisamment récente. Sous Windows, l'installation placera le répertoire bin de ce SDK dans le PATH. Vous devrez peut-être procéder manuellement à cette opération sous d'autres systèmes. Vous devrez aussi installer sur votre machine VirtualBox, le logiciel de virtualisation de Sun Microsystems. VirtualBox, dont le maniement a été présenté dans Programmez! 124 est une couche logicielle qui émule un PC physique, et qui permet de faire tourner un système d'exploitation dans ce PC virtuel. Palm WebOS a pour base le système d'exploitation Linux et est fondamentalement un Linux. Il n'est pas nécessaire de savoir manier VirtualBox pour travailler avec le SDK. Celui-ci, lors de l'installation, créera une image de Palm webOS dans votre VirtualBox. Il suffira alors de demander à VirtualBox de lancer cette image pour obtenir un émulateur de smartphone Palm parfaitement fonctionnel [Fig.1]. Toutefois, pour une raison inexplicable, seule VirtualBox 3.0.10 est supportée. Pour travailler confortablement, procurez-vous un Eclipse Ganymède (Eclipse 3.4.2) puis installez le plugin proposé par Palm. Ensuite, suivez la suggestion de Palm et installez le plugin Aptana. Les instructions pour cela sont données dans la rubrique 'Tools' du site que Palm dédie au développement et qui se trouve à <http://developer.palm.com/>. Avec tous ces outils, vous disposerez d'un IDE complet, avec éditeur de code HTML, CSS et Javascript, et des assistants de création d'applications et de scènes. Votre serveur a écrit les exemples de cet article avec ces outils. Le code est disponible sur notre site. Il est toutefois possible d'utiliser un éditeur quelconque et les outils en ligne de commande du SDK. Nous récapitulerons quelques com-



L'émulateur de smartphone Palm tourne sous VirtualBox



Ares, l'outil de développement en ligne proposé par Palm.

mandes un peu plus loin. Une autre alternative très séduisante est l'utilisation d'Ares [Fig.2]. Ares est un IDE de développement en ligne, accessible à <http://ares.palm.com/Ares/index.html>. Au moment de la rédaction de cet article, Ares est en version bêta, mais parfaitement fonctionnel. Il propose en outre une fonctionnalité de débogage très intéressante. Ares requiert malgré tout la présence de l'émulateur sous VirtualBox sur votre machine de travail. Attention, Ares requiert un navigateur Safari 4.0+, Chrome 3.0+ ou Firefox 3.5+. Ares ne fonctionnera pas avec des versions inférieures, et il ne fonctionne pas du tout avec Internet Explorer.

## 2 ANATOMIE D'UNE APPLICATION PALM WEBS

Une application Palm WebOS est constituée de trois éléments. D'abord l'application elle-même, qui est un objet avec lequel le développeur ne travaille pratiquement jamais directement. Son rôle est de gérer le cycle de vie du Stage. Vient ensuite le Stage, que l'on peut décrire comme le socle de l'application, puis les Scenes. Le Stage, parfois appelé Card (carte) est la fenêtre conteneur, fenêtre qui contient les Scenes. Le mot anglais Stage se traduisant par scène en français, traduire la terminologie Palm est incommode et générateur d'erreurs. Nous emploierons donc la terminologie anglaise. Le Stage présente une seule Scene à la fois à l'utilisateur. Mais bien sûr la majorité des applications devront comporter plusieurs Scenes. Dans ce cas, quand une Scene est présentée à l'utilisateur, elle est placée en interne sur une pile. Lorsqu'une autre Scene doit ensuite être présentée, il y a deux possibilités. Soit la première Scene est retirée de la pile, à la suite de quoi on y place la seconde, soit on place la seconde Scene directement sur la pile. Dans ce cas la première vit toujours, et lorsque l'on retirera la seconde Scene de la pile, la première réapparaîtra devant les yeux de l'utilisateur. La pile en question est de type LIFO (Last In, First Out), du moins dans son usage courant. Toutefois, il existe des fonctions permettant de dépiler une Scene pour se rendre vers une Scene quelconque. Sur le fond, l'interface utilisateur WebOS est asynchrone, comme c'est le cas sur tous les systèmes d'exploitation ou bibliothèques modernes. Par exemple Windows, ou la bibliothèque Swing de Java. Comme telle, elle dispose d'un mécanisme d'événements qui permet au programmeur d'être informé des actions de l'utilisateur et d'y réagir. Notons qu'il existe des applications multi-Stage sous webOS, mais ceci sort du cadre de cet article.

## 3 HELLO WORLD!

Créez, avec l'assistant d'Eclipse, un nouveau projet Mojo, baptisé par exemple HelloProgrammez. Les manipulations sous Ares étant similaires nous n'y reviendrons pas. L'assistant va créer une arborescence [Fig.3] qui accueillera les fichiers constitutifs de l'application. À la racine, un fichier index.html, page par défaut, rarement utilisée en pratique. On trouve encore deux fichiers de données au format JSON. L'un décrit l'application, l'autre énumère les sources constituant l'application. Le sous-répertoire 'app' mérite toute notre attention. Il contient à son tour deux sous-répertoires : assistants et views respectivement. Les noms de ces répertoires suggèrent fortement que les applications JSON répondent à l'architecture modèle-vie-contrôleur. C'est bien le cas. Le sous-répertoire 'vues' contiendra des fichiers HTML, chaque fichier HTML étant l'ossature d'une Scene. Le sous-répertoire 'assistants' contiendra des fichiers

Javascript qui animeront les Scenes et qui en définiront la logique. Lorsqu'on ajoute une Scene à une application Mojo, que ce soit avec Eclipse ou avec l'outil en ligne de commande, un de ces fichiers Javascript contenant moult fonctions est généré. Ces fonctions concernent la construction de la Scene, son initialisation (non ce n'est pas la même chose...), et les phases d'activation et de désactivation de la Scene. Notez bien en consultant notre exemple que les fichiers obéissent à une convention de nommage, sensible à la casse de surcroît. De même, les fonctions dans les fichiers assistants Javascript obéissent elles aussi à une convention de nommage. La moindre faute de frappe ici, et plus généralement dans tout le code Javascript, fera échouer votre application sans que rien ne soit signalé. Cela peut vite être agaçant, et pour éviter de s'énerver à la recherche d'un problème, on sera très rigoureux lors de l'écriture du code.

## 4 NOTIONS DE JSON

JSON, pour Javascript Object Notation est un format de données très intuitif. Un document JSON ne comprend que deux éléments structuraux : des ensembles de paires nom-valeur et des listes ordonnées de valeurs. Voici le contenu du fichier appinfo.json qui décrit notre application. Ici nous n'avons qu'un ensemble de paires nom-valeurs :

```
{
  "id": "com.programmez.helloprogrammez",
  "version": "1.0.0",
  "vendor": "Programmez",
  "type": "web",
  "main": "index.html",
  "title": "HelloProgrammez",
  "icon": "icon.png"
}
```

Modifier dans ce fichier, par exemple, la valeur icon.png du champ icon aurait pour effet qu'une nouvelle icône serait attribuée à notre application lors de son packaging. Le champ id est le plus important, car c'est à travers lui que l'on peut arrêter une application ou la désinstaller au moyen des outils en lignes de commande.

## 5 PREMIER ESSAI ET COMMANDES UTILES

Modifiez le contenu par défaut du fichier index.html par exemple comme ceci :

```
<body>
  <h2>Programmez!</h2>
```

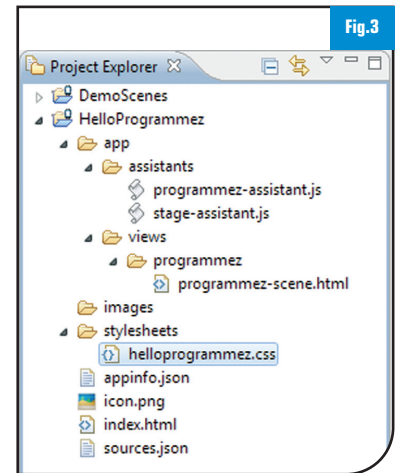


Fig.3 L'arborescence d'une application Mojo.

```
<p>
  Abonnez-vous ! :-)
</p>
</body>
```

Ensuite, lancez l'émulateur de smartphone, puis lancez l'application depuis Eclipse. Ce dernier procédera au packaging de l'application et à son installation dans le smartphone virtuel, pourvu que celui-ci soit d'abord lancé. Au bout de quelques secondes, vous verrez dans l'émulateur un résultat semblable à celui de l'illustration. [Fig.4] Comment stopper cette application sans aucun intérêt ? Le smartphone virtuel ne dispose pas, comme le Palm physique, de la zone tactile permettant d'effectuer un geste arrière. À la place, pressez la touche Escape ou la touche Home de votre clavier. Dans l'écran de défilement qui revient alors, on tue une application en l'expédiant vers le haut d'une pichenette. Il en va de même avec l'émulateur. Éjectez l'application vers le haut d'un habile et autoritaire coup de souris (oui, il faut s'entraîner un peu :-). Voici maintenant un récapitulatif de commandes en ligne qui vous seront utiles si vous travaillez en dehors d'Eclipse, ou peut-être pour intervenir brutalement en cas de problème.

Commande	Effet
palm-generate HelloProgrammez	Génère un squelette d'application Dans un répertoire de nom HelloProgrammez.
palm-generate -t new_scene -p "name=First" HelloProgrammez	Génère une Scene de nom First et l'ajoute à l'application. Vous devez être placé en amont du répertoire HelloProgrammez pour lancer la commande.
palm-package HelloProgrammez	Procède au packaging de l'application. Vous devez être placé en amont du répertoire HelloProgrammez pour lancer la commande.
Palm-install com.programmez.helloprogrammez-1.0.0.all.ipk	Installe l'application dans le smartphone réel ou virtuel. Si les deux sont présents, dans le premier trouvé. Voir la documentation pour plus de détails.
palm-install -r com.programmez.helloprogrammez	Désinstalle l'application.
palm-launch c om.programmez.helloprogrammez	Lance l'application.
palm-install -c com.programmez.helloprogrammez	Stoppe l'application.

## 6 TRAVAILLER AVEC LES WIDGETS

Il est maintenant temps d'enrichir notre application HelloWorld. Nous voulons que celle-ci présente un bouton et affiche un texte lorsque l'utilisateur clique sur le bouton, comme illustré. [Fig.5] Dans la philosophie de la programmation Palm, on ne modifiera pas le fichier index.html. On préférera créer une Scene. Faites ceci avec l'assistant d'Eclipse. Dans notre exemple, nous avons baptisé cette Scene programmez. L'assistant a généré un fichier programmez-scene.html que nous modifions comme suit :

```
<div id="main" class="palm-hasheader">
<div class="palm-header">Programmez!</div>
<div id="affichage" class="palm-body-text"></div>
<div id="MonBouton" name="MonBouton" x-mojo-element="Button"></div>
</div>
```

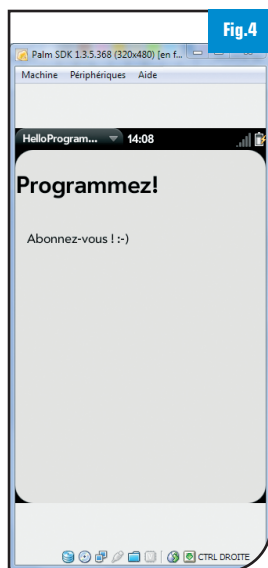
Nous constatons qu'il ne s'agit pas d'un document HTML complet, mais d'un fragment seulement. **Attention!!** L'id main du div englobant est obligatoire, si vous ne voulez pas vous arracher les cheveux. La zone d'affichage est un div tout à fait classique, affublé d'un style Palm. Vous pouvez en apprendre plus sur ce style dans la documentation, et fournir un autre style si vous le souhaitez. Il suffit pour cela de modifier le fichier helloprogramez.css dans le répertoire stylesheets. Le dernier div contient le bouton. x-mojo-element fait référence à un widget fourni par le système.

Avec notre div, l'emplacement du bouton est préparé, mais il ne s'affichera pas. Nous devons d'abord l'initialiser. Ce qui nous amène au fichier programmez-assistant.js.

Par défaut, celui-ci contient des fonctions. La première *ProgrammezAssistant* est dite constructeur, et c'est un nom qui est trompeur. Dans cette fonction on initialisera des variables ou autres, mais **ce n'est pas l'endroit** pour initialiser les widgets.

L'endroit pour cela est la fonction *ProgrammezAssistant.prototype.setup* dont voici le code :

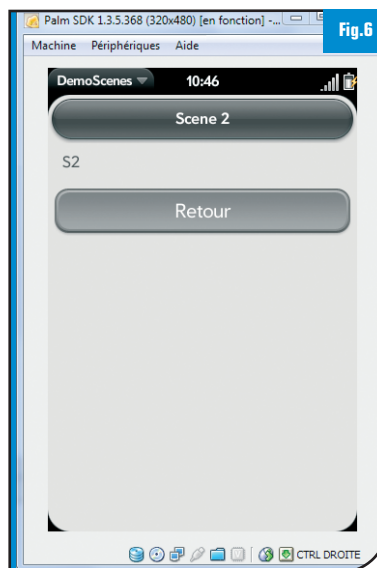
```
ProgrammezAssistant.prototype.setup = function() {
  // Remplissage du div d'affichage
```



Notre première application constituée uniquement du fichier index.html



Voici notre application enrichie de widgets.



Cliquer sur le bouton retour passera une valeur à la Scene précédemment empilée



Notre première Scene a récupéré un paramètre passé par la deuxième Scene.



```

this.controller.get("affichage").update("Cliquez le bouton");
// Instanciation du bouton
this.monbuttonAttributes = {};
this.monbuttonModel = {
  label : "Click me",
  disabled : false
};
this.controller.setupWidget("MonBouton",
  this.monbuttonAttributes, this.monbuttonModel);
// Connexion à un gestionnaire d'événement
// pour le clic sur le bouton
Mojo.Event.listen(this.controller.get("MonBouton"), Mojo.Event.tap,
  this.onClick.bind(this));
}

```

Ce code nous montre d'abord comment accéder un élément de la Scene. Ici, avec la méthode *get*, nous accédons à la zone d'affichage, puis nous en modifions le contenu. Remarquons encore que le modèle du bouton est défini avec une structure de données JSON. Enfin nous connectons le bouton à un gestionnaire d'événement. Celui-ci n'existe pour l'instant pas. Nous devons l'écrire :

```

ProgrammezAssistant.prototype.onClick = function(event) {

  this.controller.get("affichage").update("Abonnez-vous ! :-)");
}

```

Nous avons terminé pour la Scene. Mais comment celle-ci viendra-t-elle sur l'écran ? Pour cela nous devons modifier une fonction de gestion du Stage dans le fichier *stage-assistant.js*, comme ceci :

```

StageAssistant.prototype.setup = function() {
  this.controller.pushScene("programmez");
}

```

Tout simplement, par ce code, nous poussons notre Scene sur la pile, et la voici qui apparaît :-). Nous devons maintenant être très attentifs au point suivant: à chaque fois, nous avons utilisé un objet *controller*. Mais il ne s'agit pas du même selon le contexte. Le runtime en connaît trois : *AppController*, *StageController*, et *SceneController*; et fournit le controller correspondant au contexte. Si l'on a besoin d'un autre contrôleur dans un contexte donné, alors cet autre contrôleur doit être transmis au contexte. Nous allons voir comment, avec notre dernier exemple.

## 7 TRAVAIL AVEC LA PILE DE SCENE, ET PASSAGE DE PARAMÈTRES

Notre dernier exemple est composé de deux Scenes (*first* et *second*). Si dans la deuxième on clique sur le bouton retour [Fig.6], on transmet une chaîne à la première, chaîne qui sera affichée dans cette même première Scene, après dépilement de la seconde. Si l'on presse la touche *Escape* (geste arrière) alors la seconde Scene est dépilée, sans passage de paramètre. Voici les points importants du code. La première Scene contient quant à elle un bouton 'Suivant'

qui permet de passer à la deuxième Scene [Fig.7]. Tout d'abord dans *stage-assistant.js*

```

StageAssistant.prototype.setup = function() {
  this.controller.pushScene("first", "S1", this.controller);
}

```

On pousse la première Scene en passant des paramètres additionnels, une instance de *StageController* notamment. Ensuite, dans le constructeur de Scene (fichier *first-assistant.js*) nous récupérons ces paramètres, dont le *StageController* fait partie.

```

function FirstAssistant(texte, sc) {
  this.texte = texte;
  this.stageController = sc;
}

```

Dans cette première Scene, si l'utilisateur clique sur le bouton 'Suivant', on pousse la deuxième Scene, comme ceci :

```

FirstAssistant.prototype.onClick1 = function(event) {
  this.stageController.pushScene("second", this.stageController);
}

```

C'est ici que le *StageController* passé précédemment, nous est utile. Quand on quitte une Scene, une méthode *deactivate* est appelée par le Runtime. Quand on entre dans une Scene, c'est une méthode *activate*. Dans notre cas, cette dernière est appelée quand on revient à cette première Scene depuis la deuxième. Voici le code :

```

FirstAssistant.prototype.activate = function(texte_recu) {

  if (texte_recu != undefined) {
    this.texte = this.texte+texte_recu;
  }

  this.controller.get("first_affichage").update(this.texte);
}

```

Mais comme cette méthode est aussi appelée au premier affichage de la Scene, nous devons bien vérifier que le paramètre n'est pas *undefined*. Le code gérant la seconde Scene est similaire. La seule partie intéressante est la gestion du clic du bouton 'Retour':

```

SecondAssistant.prototype.onClick1 = function(event) {
  this.stageController.popScene(this.texte);
}

```

Toujours au moyen d'une instance du *StageController* qui a été passée par la Scene 1 lors de l'empilement de cette Scene 2, nous dépilons cette dernière, tout en passant un paramètre à la méthode *popScene*. Le runtime va se charger de passer ce paramètre à la méthode *activate* de la première Scene.

■ Frédéric Mazué - [fmazue@programmez.com](mailto:fmazue@programmez.com)

## WEB

## Manuel d'accessibilité du web



Difficulté : \*\*

Editeur : éditions Poonai

Auteur : Alain Gravelet

Prix : 28,70 €

Régulièrement

dans Programmez !, nous abordons les problèmes d'accessibilité des applications, des sites web malgré des spécifications comme Wai, WCAG 2, RGAA et Accessi-Web. L'accessibilité constitue un défi majeur des développeurs web qui oublient la plupart du temps ces aspects que les éditeurs cherchent à démocratiser, à rendre plus transparents pour le développeur. L'auteur revient sur les principes des spécificités disponibles sur le marché et comment un développeur peut et doit les implémenter, les utiliser. Le livre présente point par point, avec exemples, les différents éléments constituant l'accessibilité. Espérons que cette initiative se généralise enfin !

## ADMINISTRATION

## Windows PowerShell



Difficulté : \*\*

Editeur : éditions Eni

Auteur : collectif

Prix : 39 €

Vous cherchez le livre de référence sur PowerShell v1 et v2 ? Cette nouvelle édition de Windows Powershell est

LA référence en français. De la technologie .NET aux objets COM en passant par WMI et ADSI, les nombreux cas concrets d'utilisation en entreprise vous aideront à devenir plus performant dans vos tâches quotidiennes. À travers les 5 premiers chapitres, le lecteur découvrira PowerShell sous toutes ses facettes : de la simple utilisation de l'interpréteur de commandes, jusqu'aux techniques de scripting les plus avancées. Le chapitre 6 sur la technologie .NET lui montrera que l'usage de PowerShell est pratiquement sans limites et lui ouvrira une fenêtre sur le monde de la création d'interfaces graphiques avec Windows Forms et Windows Presentation Foundation (WPF). Le cha-

pitre 9 est, quant à lui, consacré aux technologies dites " de remoting " qui autorisent l'exécution de commandes ou de scripts PowerShell à distance, grâce aux nouvelles fonctionnalités de la version 2.

## JAVA

## Algorithmique et programmation en Java

3e édition



Difficulté : \*\*\*\*

Editeur : Dunod

Auteur : Vincent

Granet

Prix : 29,90 €

On touche ici au cœur même de la puissance mathématique des langages avec l'algorithme que tout développeur doit maîtriser ou du moins, connaître. L'objectif de l'auteur est de proposer des méthodes pour créer des applications valides et fiables en s'appuyant sur l'algorithme, la programmation impérative et l'orienté objet, sans oublier les notations formelles. Bref, que du bon pour le codeur exigeant. Dans cette nouvelle édition, tous les exemples, les algorithmes, les structures de données ont été mis à jour pour Java 6. En tout, on dispose de 157 exercices pour apprendre de nouvelles pratiques !

## JAVA

## Java Server Faces



Difficulté : \*\*\*

Editeur : éditions Eni

Auteur :

F-X Sennesal

Prix : 34 €

L'ouvrage s'adresse prioritairement aux développeurs d'applications web maîtrisant le langage Java. Mais il peut également intéresser les web designers qui souhaitent comprendre comment les composants JSF peuvent leur permettre de concevoir des interfaces web efficaces, sans pour autant intervenir dans le code source Java des applications. Pour cela, l'auteur propose de très nombreux exercices autour de JSF en environnement Eclipse / Tomcat. Une bonne manière de parfaire ses connaissances.

## SPÉCIAL MOBILE

## Développez vos applications pour iPhone et iPad

Difficulté : \*\*\*

Editeur : Micro Application

Auteur : Jean-Pierre Imbert

Prix : 20 €

Comment développer pour iPhone et iPad ? Si c'est une question que vous vous posez, ce livre fournira un début de réponse. De manière relativement progressive, l'auteur décrit la structure d'une application et les grands principes d'Objective-C à connaître. On y aborde les views, les enchaînements, l'enrichissement des applications, les fonctions avancées, la 3D, l'utilisation d'Interface Builder, etc. Si sur le principe, le livre remplit son contrat, on peut tout de même regretter que certaines parties ne soient qu'effleurées, comme les applications à plusieurs écrans et les implications au niveau du code et de l'interface. Mais le grand mérite de l'auteur est de permettre de démarrer en quelques heures une première application, même si le développeur ne comprend pas toutes les subtilités des outils et du langage. On peut aussi regretter l'absence de téléchargement des exemples de code.



## Programmation Android

Difficulté : \*\*\*

Editeur : Eyrolles

Auteur : collectif

Prix : 35 €



Écrit par des experts français d'Android que les lecteurs de Programmez connaissent bien, ce livre est de facto une bible pour tout développeur voulant coder sur la plate-forme Android. Très complet, il aborde : les outils, l'architecture des applications et du système, la configuration des applications, la localisation, les threads, etc. Clair et abondamment illustré de code et de bons conseils, l'ouvrage est agréable à consulter et à suivre pas à pas pour le programmeur novice ou expert. Et cerise sur le gâteau, les codes sont disponibles en téléchargement ! Une très belle réussite ! Vivement la suite...

N'ayez plus honte de faire votre site web...



REFLEXION/FAITE - © Crédits photos : Image Source.

Avec les packs WebSite,  
créez un site web élégant et performant  
en quelques clics.

DÉCOUVREZ UNE SOLUTION  
SIMPLE, RAPIDE ET INTUITIVE POUR  
CRÉER UN SUPERBE SITE WEB :  
LE VÔTRE.

- 250 modèles de sites de grande qualité
- Bibliothèque de 1000 images haute résolution
- 14 jours d'essai gratuits sans engagement
- Un seul pack qui couvre tous vos besoins (nom de domaine, hébergement, email...)
- Amen, un interlocuteur unique



DOMAINE



EMAIL



HÉBERGEMENT



SERVEUR



PUBLICITÉ



PROTECTION  
DE MARQUE



E-COMMERCE



CRÉATION  
DE SITE

 **Amen**  
A DADA COMPANY

0 892 55 66 77  
[0,34 €/mn]

[www.amen.fr](http://www.amen.fr)



Gwénaél  
Responsable Réseaux et Télécom

Isabelle  
Directrice d'Agence

Alain  
Responsable Commercial



Communiquer  
et partager les idées  
en toute sécurité

Parfois, l'information que l'on recherche est sur un disque dur, parfois elle se trouve dans la tête de quelqu'un. En déployant Microsoft Exchange Server, avec ses services de Communications Unifiées, SharePoint, et ForeFront, vous ferez des économies en permettant à vos collaborateurs d'échanger des idées où qu'ils soient et d'accéder aux informations dont ils ont besoin en toute sécurité.

Pour une communication et une collaboration plus efficace, plus sûre et plus facile à gérer, rendez-vous sur : [www.nouvelle-efficacite.fr](http://www.nouvelle-efficacite.fr)