

Tout savoir sur **TEAM FOUNDATION SERVER 2010**

Supplément détachable 24 pages

Microsoft®  
**Visual Studio® 2010**

Le magazine du développement

PROGRAMMEZ !

# PROgrammmez !

www.programmez.com

mensuel n°138 - février 2011

Développer pour les processeurs

# ATOM



**Intel AppUp** : votre AppStore multiplateforme

Optimiser et déployer son application

**TechDays 2011**

**Cloud,  
Mobile  
et ALM**

**Cloud**

Déployez ASP.net  
sur Windows Azure

**Smartphone**

Faire cohabiter  
Flex et Android

**GreenIT**

Utiliser  
le gestionnaire  
d'énergie  
de Windows

Choisir sa plateforme de



Blogger, Dotclear, Wordpress...

# BLOG

**SGBD**

# NoSQL :

révolution ou simple buzz ?

M 04319 - 138 - 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

N°1 EN FRANCE

WINDEV®

- Crée des applications Windows, Linux, Mac,... compatibles Internet, Intranet, Android, Windows Phone 7
- Éditeur d'analyses (UML, Merise...)
- RAD (patterns)
- Lien avec toutes les bases de données (ODBC, OLE DB, Lien natif), Oracle, SQL Server, AS/400, Informix, DB2, MySQL, PostgreSQL, SQLite,...
- HyperFileSQL: puissante base de données Client/Serveur gratuite incluse
- Générateur d'états PDF, Codesbarres
- Accès natif SAP R/3, Lotus Notes, ...
- Gestion de planning, des Exigences
- Audit automatique
- Langage de 5<sup>e</sup> génération (L5G)
- SNMP, Bluetooth, TAPI, OPC, FTP, HTTP, Socket, Twain, API, DLL, XML...
- Domotique • Liaisons série et USB
- Débogage à distance
- Profiler, Refactoring
- Multilangue (64 langues)
- Installateur 1-clic & Push
- Gestionnaire de versions
- Gestion de l'infrastructure Logicielle
- Tout en français
- etc...

16

CRÉEZ VOS  
APPLICATIONS  
POUR PC, MAC,  
LINUX, INTERNET,  
MOBILES,...

VOTRE CODE EST  
MULTI-PLATEFORMES :  
Windows, .Net, Java, PHP, Linux,  
Mac, J2EE, XML, Internet, SaaS,  
Windows Phone 7, Pocket PC,  
Android, ...

DÉVELOPPEZ

10 FOIS PLUS VITE

Elu «Langage le  
plus productif du  
marché»

VERSION  
EXPRESS  
GRATUITE  
Téléchargez-la !



996 NOUVEAUTÉS

Fournisseur Officiel de la  
Préparation Olympique



www.pcsoft.fr

DEMANDEZ LE DOSSIER GRATUIT

► Dossier gratuit 260 pages sur simple demande. Tél: 04.67.032.032 info@pcsoft.fr



## \\ actus

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

## \\ evenement

TechDays 2011 : du Cloud, du Mobile, de l'ALM .....	10
VisualStudio 2010 Service Pack 1 : quoi de neuf ? .....	12

## \\ matériel

Test du Nas MyDitto .....	15
---------------------------	----

## \\ webmaster

### Choisir sa plateforme de **blog** !

Blog ou CMS, que choisir ? .....	19
Promis, aujourd'hui, je trouve ma plateforme de blog ! .....	20
WordPress, le CMS sur mesure .....	22

## \\ sgbd

### Les bases NoSQL vont-elles tuer les bases relationnelles ?

Vers la fin du tout relationnel ! .....	27
NoSQL dans un contexte Windows Azure .....	33
La mort des bases de données relationnelles et du SQL ? .....	36
Prévalence : le meilleur de NoSQL .....	40

## Tout savoir sur Team Foundation Server 2010

Cahier détachable folioté de 1 à 24  
Sommaire en page 2 du cahier

## \\ dossier

### Atom, AppUp : une nouvelle plateforme pour le développeur !

Tout ce que vous avez toujours voulu savoir sur AppUp .....	71
Un spécialiste du livre numérique se met à AppUp .....	74
Développement de logiciels pour la plateforme Intel Atom .....	75

## \\ carrière

Enquête : l'informatique booste la carrière des femmes .....	82
--	----

## \\ code

Créer votre première application ASP.NET MVC 3 sur Windows Azure ....	83
Combiner Flex et Android c'est possible ! .....	88
CKlone : l'IDE du Cloud ! .....	93
Ajuster la fréquence de votre processeur avec l'API Windows Power Management .....	97
Ecrire des extensions pour les navigateurs Opera et Safari .....	102

## \\ temps libre

Les livres du mois .....	106
--------------------------	-----



6



10



19

overblog



71



88

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.



**NOKIA**  
Connecting People

# Chaque jour, des millions d'applis Nokia

sont téléchargées dans  
le monde\*

## A quand la vôtre ?

Développer une appli Nokia n'a jamais été aussi simple et avantageux. Rapprochez-vous de nos **4 millions d'utilisateurs de Smartphones en France** et accédez instantanément à plus de **190 marchés à travers le monde** grâce à Ovi Store.

\* 3,5 millions d'applications téléchargées par jour sur Ovi Store au 01/01/2011



Rendez-vous sur  
[www.nokia.fr/develop](http://www.nokia.fr/develop)





## Que ma boutique est belle !

Le 2e effet App Store d'Apple existe-t-il ? Depuis que le téléphone de la pomme croquée est sur le marché, la stratégie de la boutique en ligne - notre fameux App Store - a démontré sa pertinence et son incroyable succès. Conséquence ? Tout le monde copie le modèle Apple. Eh oui, la concurrence a beau dire « bouh le iPhone est un méchant téléphone », il n'empêche que constructeurs et opérateurs se rendent compte que Apple a bouleversé la consommation des applications mobiles. Aujourd'hui, deux boutiques dominent le marché : App Store et Android Market. Le marketplace de Windows Phone 7 doit encore prouver sa pertinence et consolider les bonnes impressions autour du mobile Microsoft. Sans oublier, celui de Nokia, Ovi Store qui ne cesse d'améliorer sa présence et le nombre d'applications disponibles.

Mais l'histoire ne s'arrête pas là. Apple cherche maintenant à faire la même révolution applicative sur le desktop avec le Mac App Store, en reprenant le même modèle que la boutique iOS. Mais si sur un téléphone il est naturel d'acheter et d'installer une application en un doigt (euh oui difficile de parler de clic), sur le desktop, cette approche n'est pas courante. Et le Mac App Store promet de faciliter la vie des utilisateurs : installation en un clic ! Cela change le paradigme et nous ne pouvons que nous en réjouir. Fini les installations de logiciels nécessitant 10 clics, 30 à 60 minutes d'installation ! Non ce n'est plus possible. Bref, il était logique que Apple tente de casser le modèle applicatif. Comme d'habitude, cela a fait rire les concurrents, éditeurs et les anti-Apple. Mais comme pour App Store, ils vont déchanter.

Pour le développeur, il s'agit là encore d'une nouvelle opportunité. Bien entendu, il faut adapter son code, prendre en compte de nouvelles API mais au final, le développeur retrouve la liberté qu'il a dans le développement sur mobile. Et comme vous le verrez dans notre dossier Atom / AppUp, la concurrence investit dans ces boutiques. Microsoft devrait logiquement faire de même, Google est d'ores et déjà sur le pied de guerre pour son futur ChromeOS.

Côté technique, ces changements ne sont pas colossaux : quelques API, un nouveau modèle de déploiement à considérer, établir une stratégie de vente (si l'application est payante), respecter les règles, même si finalement, le développeur sera toujours inquiet des politiques d'approbation des éditeurs. Que l'on soit d'accord ou non avec ces pratiques, faudrait-il réellement laisser toute liberté au développeur, au risque de laisser la plateforme devenir hétérogène quant à la qualité des applications ? L'avenir nous le dira. Et à terme, nous aurons un rapprochement des modèles applicatifs entre mobile et desktop, Mac App Store et le futur MacOS X Lion en dessinent les contours.

Cependant, le revers de la médaille est la multiplication de ces boutiques. La majorité des développeurs n'a pas les moyens d'aller sur toutes les plateformes, car cela nécessite d'adapter le code, l'interface, d'apprendre un nouveau langage. Sans compter l'apprentissage des outils de développement.

Une autre évolution pointe à l'horizon, l'usage du cloud computing dans les applications. Cette intervention se fait beaucoup sur les capacités de stockage du cloud, d'ailleurs, Microsoft met en avant le couple Windows Phone 7 et Azure Storage. Google ne dit pas autre chose avec ChromeOS.

■ François Tonic

**Editeur :** 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. Mazué, S. Saurel, O. Pavie **Experts :** J. Dollon, C. Villeneuve, P. Couzy, B. Guinebertière, O. Malassi, JP. Gouigoux, J. Horton, T. Jaskula, A. Vannieuwenhuyze, F. Queudret,  
**Illustrations couverture :** source Intel D.R.

**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

**L'INFO PERMANENTE**  
**WWW.PROGRAMMEZ.COM**



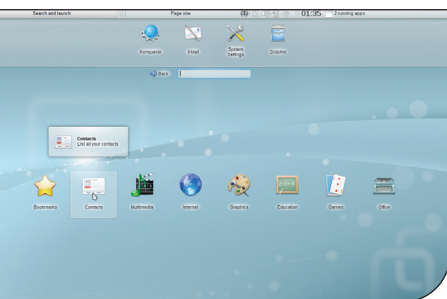
**PROCHAIN NUMÉRO**  
**N°139 mars 2011**  
**parution 26 février**

- ✓ **Cloud Computing :**  
Comment intégrer des services cloud ?  
Quelle plateforme utiliser ?  
Quels modèles de développement ?
- ✓ **CPU - GPU**  
Où et pourquoi utiliser l'accélération matérielle dans son code, son site web ?

■ **OpenJDK** a dévoilé officiellement les premiers travaux autour de la version MacOS X depuis l'arrivée d'Apple dans le projet. L'objectif reste identique aux autres versions : fournir une JDK 7 de qualité, passer la certification, assurer les performances. Cette version fonctionnera uniquement sur des Mac Intel et à partir d'un système 10.6 (pour l'instant).

■ **Mandriva** sort officiellement la version 2010.2 de sa distribution. Parmi les nouveautés, le 'Smart Desktop' qui permet l'accès dynamique à tous ses fichiers en les étiquetant, en notant ses photos, ses documents, ses mails ou ses vidéos, et en permettant de gérer ses données de façon complètement personnalisée.

■ **Modeliosoft** et Objecteering Software annoncent leur fusion dans une unique société. Modelio a été conçu comme une nouvelle génération d'atelier de modélisation, tout en reprenant l'expérience et les points forts de l'atelier Objecteering qui est une référence sur le marché depuis 1991.

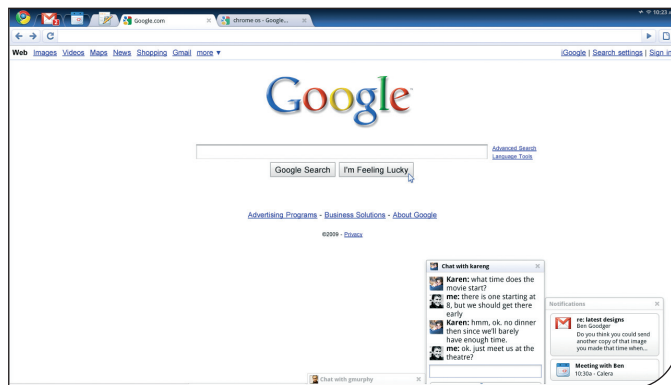


■ **KDE** dévoile une version 4.5.4 sur... Windows. Une sortie doublement importante, car outre le fait d'apporter KDE 4.5.4 sur Windows, cela témoigne d'une remise en piste du projet KDE pour Windows, après "un long travail de fond". Cette version fixe des bugs de la version précédente et supporte deux nouveaux compilateurs, dont celui de Microsoft en version 2010.

## Système Le futur d'Android se précise

La future version 3, branche indépendante de la 2.x, se précise enfin. Beaucoup de rumeurs, peu d'informations fiables. Tout d'abord, la v3 fonctionnera sur smartphone et tablette. Et la plateforme technique ne se limitera pas à celle de Nvidia Tegra 2. Android 3, alias Honeycomb, propose un copieux menu : multitâche, notification améliorée, interface et ergonomie revues, nouvelle fonction de synchronisation, navigateur également largement modifié pour simplifier l'usage au quotidien, nouveau clavier, meilleure intégration des Google Apps, etc. Les nouveautés de cette v3 sont nombreuses et réjouiront les utilisateurs. Par contre, il faudra s'attendre à des modifications non négligeables dans le SDK et les API. Les développeurs devront sans doute revoir partiellement leur code ou tout du moins l'interface.

A l'heure actuelle, nous sommes en Android 2.3 (Gingerbread). Cette version améliore la partie jeu avec EGL / OpenGL ES optimisée et un nouveau framework pour la gestion du cycle de vie et des fenêtres. Côté média, l'enrichissement est présent également : support des formats VP8 et WebM



ou encore AAC. On dispose de nouveaux effets. Pour la communication, quelques nouveautés bienvenues avec le support de la caméra de face, du SIP/VoIP. Le runtime Dalvik a lui aussi subi quelques modifications : ramasse-miettes concurrent, OpenSSL 1.0.0a, JIT plus performant, nouvelles API réseaux, mise à jour du JDBC. On bénéficie aussi de belles nouveautés et améliorations sur les fonctions gyroscopiques et d'accélération. Cela devrait améliorer la gestion de l'orientation, des mouvements du terminal, ce qui est crucial pour les jeux et pour certaines applications. N'oubliez pas de prendre les dernières versions des SDK et outils pour la v2.3 : SDK Tools r8, USB Driver for Windows r4, Android NDK r5 et ADT 8.0.1. Et Google a déjà dévoilé le successeur (v2.4 ? ou 3.x ?) :

Android Ice Cream Sandwich, qui pourra sortir dès cet été.  
Site : <http://developer.android.com>

Concernant ChromeOS, aucune date de sortie précise n'est pour le moment connue. Nous restons sur « début 2011 » bien que les premiers netbooks en test aient été livrés aux Etats Unis en décembre dernier. Cependant, une version Release Candidate (RC), dernière phase avant la version finale avait été livrée courant octobre. Et le grand salon CES 2011 de début janvier n'a fourni aucune nouvelle ou annonce autour de ChromeOS. Mais l'éditeur ne répond pas non plus à une question importante : ChromeOS est-il mort-né, doit-il fusionner avec Android ? De cette réponse dépendra sans doute une partie de l'avenir de ChromeOS...

## agenda \

### FEVRIER

• Du 08 au 10 février, Palais des Congrès de Paris : **TechDays 2011**  
Rendez-vous incontournable des développeurs, décideurs et professionnels de l'informatique, Les Microsoft TechDays sont une occasion unique de se former sur toutes les nouveautés Microsoft et de découvrir les tendances du marché.  
[http://www.microsoft.com/france/mstechdays/?CR\\_CC=200012242](http://www.microsoft.com/france/mstechdays/?CR_CC=200012242)

Le mardi 22 février 2011, en ligne, 14h30

Webinar organisé par Zend Technologies : **Développez, gérez et scalez vos applications dans le Cloud**. Présentation de la nouvelle Zend PHP Cloud Application Platform, qui vous aide à développer, gérer et scaler vos applications PHP critiques dans le cloud.  
<http://www.zend.com>

### MARS

• Du 02 au 05 mars 2011, Cité Universitaire Internationale 17, bd Jauréguian 75014 Paris,  
**Conférences Symfony Live 2011**  
Sensio Labs annonce ses conférences Symfony pour "évangéliser autour du framework, et plus largement autour de la professionnalisation de PHP et la dissémination des bonnes pratiques de développement."

<http://www.symfony-live.com/paris/schedule>

• Du 29 au 31 mars 2011, Paris Porte de Versailles, **Machine to Machine 2011**.

L'édition 2011 du Salon est le Rendez-vous de toutes les entreprises qui souhaitent découvrir les nouvelles technologies et applications développées dans ce domaine par les principaux acteurs du marché MtoM (M2M)  
[www.salons-solutions-electroniques.com](http://www.salons-solutions-electroniques.com)



# DEVELOPPEMENT

Applications de Business Intelligence riches en WPF et Silverlight



Nos grilles de pivot pour WPF et Silverlight permettent à vos utilisateurs d'analyser leurs données et prendre des décisions clés. Visitez [infragistics.com](http://infragistics.com) pour les essayer dès aujourd'hui!



**NetAdvantage<sup>®</sup>**  
for Silverlight Data Visualization

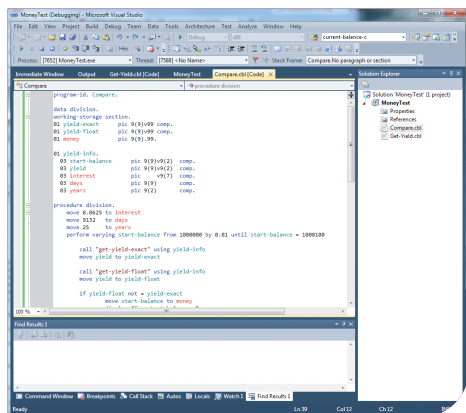


**NetAdvantage<sup>®</sup>**  
for WPF Data Visualization

**Infragistics<sup>®</sup>**

Infragistics Ventes France   
0800 667 307 • Infragistics Europe Ventes  
+44 (0) 800 298 9055 • [@infragistics](http://t.infragistics)

■ **Micro Focus dévoile Visual Cobol R3.** Cet outil permet de développer des applications Cobol avec Visual Studio et Eclipse pour Windows, .Net, le cloud computing (Azure), Linux et toute JVM. Il s'agit d'une version majeure pour l'éditeur afin d'améliorer la productivité du développeur Cobol tout en apportant le support des dernières technologies.



■ **ASP.Net MVC 3.0** est désormais disponible. Une mouture qui se distingue notamment par Razor Intellisense, c'est-à-dire l'intégration dans Visual Studio et Visual Web Developer Express de Razor, le nouveau moteur de templates, associé à la coloration syntaxique et l'achèvement de code. C'est aussi l'intégration du packager open source NuGet qui permet de facilement délivrer des applications faisant appel à des librairies tierces. Et c'est encore un meilleur support du Javascript.

■ **Un bug dans PHP 5.3 ?** Rick Regan donne à penser qu'un bug obscur de PHP 5.3 (et peut-être PHP 5.2) pourrait servir de vecteur pour une attaque de déni de service de manière extrêmement simple. PHP va s'égaler dans une boucle infinie par cette seule instruction : `<?php $d = 2.2250738585072011e-308; ?>` Tous les détails ici : <http://www.exploringbinary.com/php-hangs-on-numeric-value-2-2250738585072011e-308/>

■ **Ubuntu** va abandonner la suite bureautique OpenOffice au profit de la nouvelle communauté LibreOffice. Ce changement sera effectif dès Ubuntu 11.04. La décision ayant été motivée, partiellement du moins, par le fait que le package officiel de OOo demeurait en 3.2.1 et tardait à passer en version 3.3 comme LibreOffice et Oracle Open Office (version payante). Il existe désormais un décalage entre la version payante et l'édition open source, OpenOffice.org dont la v3.3 est encore en pré-version. Site : <http://www.openoffice.org/>

## Web : guerre sur les formats vidéo

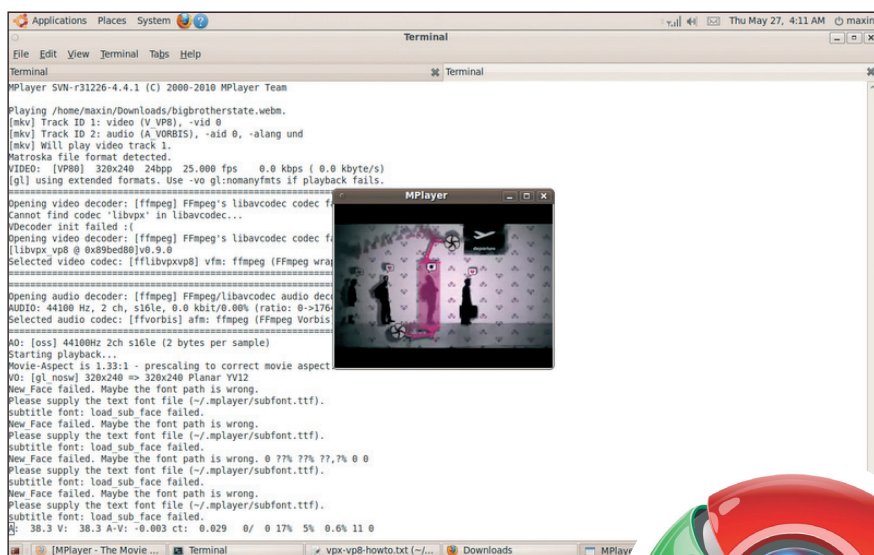
Mi-janvier, Google annonce, via son blog officiel, l'abandon prochain, du format vidéo H.264, au profit de deux formats ouverts : WebM (VP8) et Theora pour la balise vidéo HTML 5 de Chrome. H.264 est réputé pour sa qualité, et ses bonnes performances.

Et Google reconnaît ainsi l'importance du codec vidéo sur Internet mais préfère tout de même le retirer dans ses logiciels. En effet, H.264 sera entièrement supprimé et plus aucune ressource interne dédiée à ce format. N'est-ce pas là une volonté de couler, sans le dire, le codec ? Car pour les observateurs, ce retrait de Google porte un mauvais coup à H.264. Google met en avant les formats ouverts (initiative louable) car H.264 possède une licence et des royalties (même si cela a été réglé depuis de nombreux mois). Microsoft (qui participe activement au codec) a réagi, sous la

tant plus que Firefox (Microsoft a dévoilé un plug-in H.264 en décembre dernier pour ce navigateur), Safari, IE, Flash supportent ce codec.

En regardant de plus près la situation, une certaine confusion règne. Car, Google pousse un autre argument, stratégique : rendre disponible une extension webM pour Safari et IE 9 (projet développé par le WebM Project Team), même si aucune date n'est fournie. Mais reste à savoir si ces extensions seront fonctionnellement au niveau du support fourni dans Chrome ?

Rappelons que WebM est un format de



forme d'un billet inspiré comparant les codecs vidéo à des langues, évoquant un changement de langues aux Etats-Unis : remplacer l'anglais (H.264) par l'espéranto et le klingon...

Il faut dire que cette affaire n'est pas neutre car H.264 est très populaire et largement diffusé quel que soit le système, le navigateur et surtout les outils d'encodage. Et si Google prône les formats ouverts, est-ce vraiment une bonne nouvelle pour l'utilisateur final qui risque dans les prochains mois de se retrouver devant des vidéos non consultables sous Chrome. Surtout, on multiplie une fois de plus les codecs. Et Google aurait pu garder le support « mort » de H.264, d'au-

media ouvert et sans royalties. Il contient des flux audio et vidéo basés sur les codec VP8 et Vorbis. La structure des fichiers WebM s'appuie sur le conteneur Matroska. Celui-ci définit la structure du fichier WebM qu'il faut voir comme une enveloppe. Dans le même esprit, Google pousse un autre format, WebP. Il s'agit d'un format image utilisant les mêmes principes que VP8. Il se veut le successeur de JPEG et d'autres formats aujourd'hui très largement utilisés sur le web.

La suite dans le prochain numéro ?





## Environnement

### Microsoft WebMatrix en version finale



C'est le 13 janvier dernier que Microsoft a annoncé la version finale de sa plateforme web : WebMatrix. Jusqu'à présent disponible en pré-version, WebMatrix permet de créer, de déployer rapidement un environnement web pour utiliser et installer un CMS, un blog, un site web. Le but est de simplifier au maximum la phase d'installation, via un assistant. Microsoft souhaite donner aux développeurs, experts ou pas, le moyen de créer rapidement leur blog, leur site avec Drupal ou d'utiliser un panier Paypal, de créer des liens avec facebook, etc.

Pour faciliter le développement, WebMatrix arrive avec une nou-

velle syntaxe Razor ressemblant beaucoup à HTML et des helpers, permettant de coder très rapidement du code paypal, par exemple. WebMatrix embarque IIS Express, ASP.NET Web Page, SQL Server Compact et est disponible en 9 langues !

Aujourd'hui, 40 outils open source sont partenaires de WebMatrix. Plusieurs hébergeurs ont d'ores et déjà annoncé le support des déploiements WebMatrix, en France : Ikoula et Netissime. Il se murmure que dans une future version, le support de Windows Azure serait disponible par défaut (avec Azure Storage pour le stockage des données).

## Cloud

### AMAZON AWS MET À JOUR EC2

Amazon EC2 évolue chaque mois. En janvier, Amazon annonce plusieurs nouveautés. Parmi les plus intéressantes : VM Import pour EC2, ou comment importer des images de machines virtuelles existantes dans son EC2. La fonction propose un mécanisme, le plus automatique possible. Elle supporte : VMware VMDK pour Windows Server 2008 R2, d'autres supports sont prévus à court terme ainsi que des formats d'images différents (Xen, Red Hat, VirtualBox, Hyper-V, etc.). Nouveau support premium et des prix en baisse pour le support existant. Le support premium est annoncé comme 24x7x365, avec un accès aux ingénieurs. D'autre part, Oracle s'associe avec Amazon pour être présent sur EC2. Et la liste des solutions Oracle capable de fonctionner sur EC2 est longue : Oracle 11g, Oracle Fusion Middleware, Oracle Enterprise Manager. Surtout, les licences achetées sont compatibles avec EC2, sans complément. Pour faciliter le travail, Oracle fournit des images AMI d'Amazon.

# Passionné(e) par le développement et les nouvelles technologies Microsoft ?

Spécialisez-vous chez **Codit**, leader en intégration, et devenez expert en Microsoft BizTalk, AppFabric et Azure !

Rencontrez-nous aux TechDays\* et gagnez votre CDI d'expert et/ou votre propre Windows Phone.

\* village DEV, porte Maillot,  
les 8, 9 et 10 février 2011

**codit**  
integrating your business

11, Rue de l'Escaut  
75019 Paris,  
France  
jobs@codit.eu  
www.codit.eu

# TechDays 2011 : du cloud, du mobile, de l'ALM

Du 8 au 10 février, Microsoft fera son show annuel à Paris : les TechDays. Sur trois jours, plus de 17 000 visiteurs (dont 11 000 uniques) sont attendus pour assister aux trois plénières, aux 332 sessions techniques, rencontrer et visiter les 150 communautés et partenaires présents. Cette année, le cloud computing, et particulièrement Windows Azure, sera la vedette de l'événement. Le message est clair : passez au cloud computing !

L'édition 2011 offre un intérêt tout particulier pour les utilisateurs et les partenaires, comme nous l'a confirmé Laurent Ellerbach (Microsoft France), tous les stands ont été vendus dès décembre 2010, ce qui prouve l'importance de l'événement. La zone cloud computing sera particulièrement vaste cette année, contrairement à l'an dernier.

## Tout pour le cloud

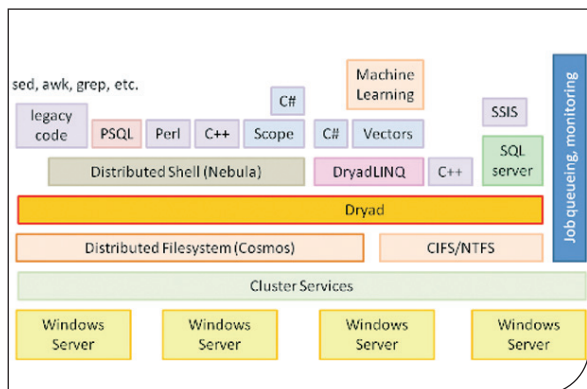
Le cloud computing et Windows Azure seront les grandes vedettes de

l'édition 2011. Et pour cause, en 2010, nous n'étions qu'au début du phénomène, depuis un an, les choses ont beaucoup bougé et Microsoft a peaufiné les outils, son offre Online Services ainsi que la partie IaaS avec le nouveau Hyper-V Cloud.

Le cloud sera présent un peu partout. Sur Windows Azure uniquement, 12 sessions sont prévues. Cela va de la présentation de la plateforme, au support des langages .Net (PHP et Java), à la migration des applications, l'administration Azure ou

encore comment concilier Windows HPC et Windows Azure pour les calculs intensifs ! La partie Données n'est pas oubliée avec OData qui aura droit à plusieurs sessions et bien entendu, SQL Azure !

D'autres sujets passionneront les développeurs : le multicore et les tests. Sur l'ALM, plusieurs sessions se concentreront sur Team Foundation Server 2010. Bien entendu, vous retrouverez les sessions purement développeurs pour mettre les mains dans la mécanique !



## Les technologies et outils 2011 -

S'il faut toujours garder une oreille et un œil ouverts dans les allées, c'est bien pour dénicher quelques indiscretions de projets internes, de nouveaux outils qui apparaîtront dans les 18 prochains mois...

**Surface 2.0 :** la table Surface refait surface dans une version 2 très attendue. Les équipes ont travaillé dur pour retailler le projet, le booster, l'améliorer. Au final, un matériel bien plus fin, un écran haute définition, une expérience et une navigation 100 % tactile, terminé le clavier pour certaines fonctions ou configurations... Le système sera plus réactif et moins sensible aux variations de lumière et surtout, il sera plus personnalisable que la v1. Une autre bonne nouvelle est l'administration à distance via Powershell ! Et le modèle de développement a été remis à jour (portabilité avec le multitouch de Windows 7). La table embarque un Windows 7 64-bit,

.Net 4, WPF 4, XNA Framework 4. Le tout est propulsé par un matériel provenant de Samsung (SUR40). Pour un prix (un peu plus) abordable...

**Visual Studio Async :** Cette nouvelle version intégrera la programmation asynchrone au cœur des applications .Net. Cette approche est, selon Microsoft, la prochaine grande évolution de la programmation afin de répondre aux problèmes de montée en charge, de réactivité de l'interface et de latence dans les applications. F# a aussi droit à sa session, surtout depuis son passage en licence Apache qui devrait aider à son utilisation.





### Windows Phone 7 : l'autre gros morceau

Microsoft veut réinvestir le téléphone mobile et particulièrement le smartphone. Avec Windows Phone 7 l'éditeur possède une nouvelle arme pour espérer contrer Google, Samsung, Apple, Blackberry. Disponible depuis octobre 2010, cette version constitue une remise à plat du système mobile, du modèle de développement. Les TechDays seront l'occasion parfaite de mieux comprendre Windows Phone 7, les outils, son écosystème, comment

développer dessus, profiter du marketplace, etc. Au total, une quinzaine de sessions seront jouées durant ces trois jours, avec l'embarras du choix : maîtriser Metro, consommer des services et données Windows Azure, migrer d'Android ou iOS vers Windows Phone, les problématiques d'entreprise. C'est aussi courant février que Windows Phone 7 sera mis à jour avec le Service Pack 1 (nom de code : Noddy) que l'on devrait découvrir durant les TechDays. Cette mise à jour, très attendue, apportera des fonctions

absentes jusqu'à présente comme le copier-coller. Les applications devraient connaître de réels progrès de performance, un meilleur basculement entre les applications, et le Marketplace bénéficiera d'une nouvelle navigation. Le multitâche devrait être amélioré ainsi que le support de Flash. En principe tous les utilisateurs de Windows Phone 7 pourront mettre à jour directement leur terminal mobile sans attendre le constructeur... Et ce n'est qu'un début car Windows Phone 7 aura droit dans quelques mois à une seconde (grosse) mise à jour : Mango. Cette version apporterait deux grosses nouveautés : HTML 5 et Silverlight ! Rien de moins... Et cerise sur le gâteau, une apparition ( ? ) de Internet Explorer 9... Ca va chauffer chez les développeurs en 2011 !

## - 2012 à suivre chez Microsoft

### Du développement HPC / multicore :

Microsoft lève un peu le voile sur la technologie Dryad et DryadLink grâce à deux pré-versions. L'objectif premier de Dryad est d'optimiser, d'améliorer les calculs intensifs en contexte multicore et distribué. Il se dédie donc plutôt au monde HPC. La présence de DryadLink, donc orienté données, permettra de manipuler plus facilement les données et les bases de données. Les concurrents sont Hadoop et MapReduce.

### Windows 8 se dévoile (un peu) :

Microsoft a annoncé lors du grand salon CES aux Etats Unis que le prochain Windows (Windows 8) supportera

les architectures à processeur ARM et le « system on a chip » (SoC) de différents fondeurs. L'approche SoC est adaptée aux matériels ayant peu de ressources, peu de puissance. Cela signifie que Windows sera multi-architecture et pourra donc fonctionner un peu partout (ou presque) et notamment sur les fameuses tablettes tactiles. D'autres rumeurs concernent un nouveau modèle de développement : Jupiter. Un des objectifs sera de simplifier le package applicatif et son déploiement (à l'image du Mac App Store d'Apple ?). Windows 8 n'est pas attendu avant 12 à 18 mois...

**WebMatrix :** WebMatrix a

été construit afin de rationaliser le développement Web tout en le facilitant. Si vous débutez ou vous aimez tout simplement les choses bien faites, celui-ci est fait pour vous. Il vous permet de vous focaliser sur ce qui est important : votre site Web.

**Windows Azure :** le nouveau portail Silverlight est disponible depuis fin décembre à tous les utilisateurs Windows Azure ! Et c'est un réel changement même si SQL Azure et AppFabric ne migreront que courant 2011. Et cette bonne nouvelle arrive avec de nombreuses fonctions d'administration, des guides, de la documentation et plusieurs fonctions actuellement

en pré-version (VM Role, Windows Azure Connect).

### SQL Server 2011

« Denali » : depuis novembre, les premières versions techniques (CTP) sont disponibles. La prochaine SQL Server sera disponible fin 2011. Les principales caractéristiques sont : amélioration des aspects critiques (haute disponibilité et montée en charge pour obtenir un coût de possession réduit), nouveaux outils de développement, nouvelles fonctions de BI, une administration plus automatique, intégration de données améliorée : le but, avoir la bonne donnée au bon moment pour le bon utilisateur.

# Visual Studio 2010

## Service Pack 1 : quoi de neuf ?

L'objectif de cet article est de vous présenter les nouveautés de VS 2010 introduites par le service pack 1 et dans le Feature Pack 2 ainsi que quelques bêtas de framework qui font grand bruit. Enjoy !

**L**e SP1 fait environ 500Mo et met un certain temps à s'installer ! Pour ma part j'ai eu quelques petites difficultés avec certains processus de ma machine mais rien de bien méchant.

Voici les principales nouveautés :

- **SharePoint :**  
IntelliTrace pour les projets SharePoint (et les projets .NET 64 bits)
- **Silverlight**  
Apparition du Performance Wizard for Silverlight
- **ADO.NET**  
Mode Code First pour Entity Framework
- **Tests :**  
Intégration des tests unitaires pour le framework 3.5  
Support des tests UI pour Silverlight
- **ASP.NET :**  
Installation de la dernière version d'ASP.NET MVC (MVC 3 RC2)  
IIS Express
- **RIA Services :**  
Installation du SP1 de WCF RIA Services
- **C++ :**  
Support du GPU pour les applications C++
- **Nouvelle aide, le Help Viewer**
- **Asynchrone**  
Installation de Visual Studio Async CTP  
RX Extensions
- **Génération de code à partir de l'UML**
- **Nouveau portail d'Azure**

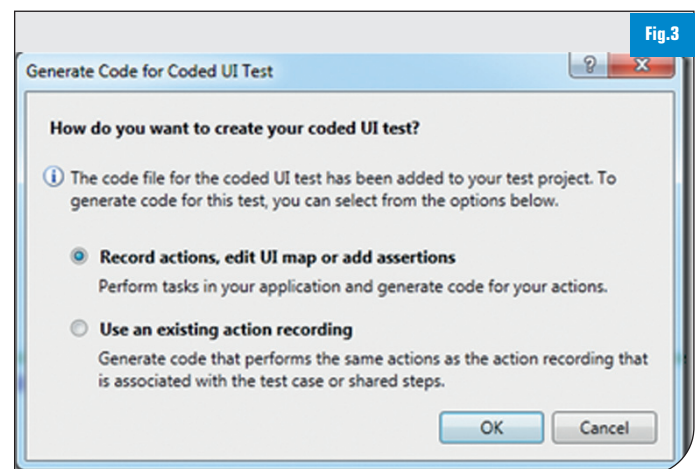
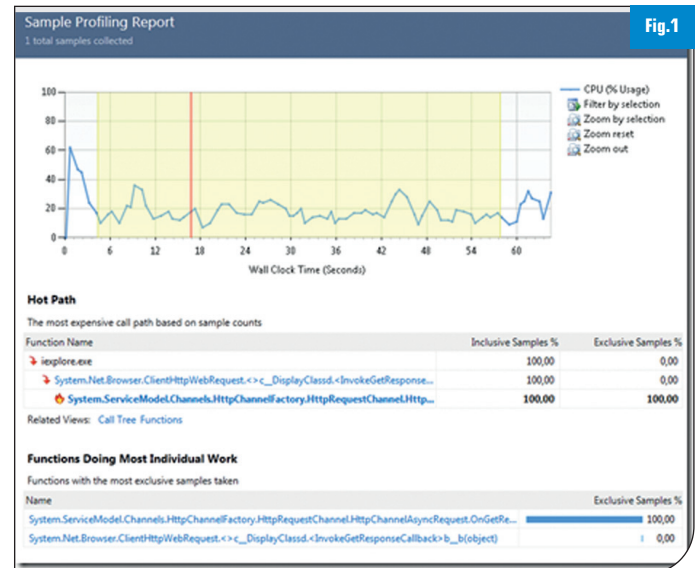
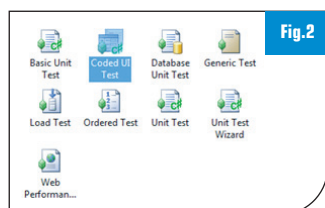
### Silverlight : performance Wizard

Le performance wizard va nous permettre de déterminer où se trouvent les problèmes de performances de nos applications Silverlight. Après le lancement de l'application, l'assistant de performances va enregistrer toutes nos actions associées à l'occupation CPU [Fig.1].

### Silverlight : tests d'interfaces

Il est possible depuis quelque temps de faire des tests d'interfaces qui consistent à enregistrer un scénario et de pouvoir le rejouer grâce à l'automation. Ceci était possible pour de nombreuses technologies (WPF, ASP.NET mais avec un autre type de tests) mais pas Silverlight! Problème résolu ! Après la création d'un projet Silverlight,

il faut ajouter un projet de Test. Et créer un test de type Coded UI : [Fig.2]. Puis nous enregistrons le scénario avec les actions à réaliser : [Fig.3]. Attention, il faut penser à ajouter une DLL spécifique à notre projet Silverlight.



### RIA Services : les nouveautés

J'ai déjà beaucoup expliqué de notions dans mon livre sur Visual Studio. RIA Services simplifie grandement la création de service MAIS pour moi cette technologie est encore trop jeune pour que je puisse la conseiller dans un environnement de production. Que ce soit parce que les relations n à n ne sont pas supportées ou pour les quelques petits bugs du même style que j'ai pu rencontrer (it's not a bug, it's a feature !). J'attendais vraiment le SP1 car le créateur de RIA Services (Brad Adams) a démissionné après la sortie de la V1 de ce produit et j'étais curieux de savoir quelle route allait prendre ce framework.



Résultat : le SP1 montre que ce projet continue, mais je trouve ce SP1 un peu léger, j'attendais encore plein d'autres fonctionnalités :

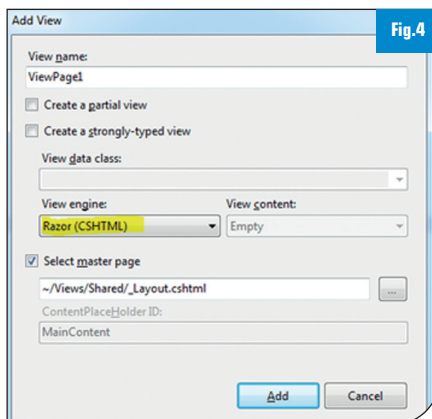
- Support des objets complexes: les classes contenant des objets type complexe peuvent maintenant être sérialisées
- Shared Entity: plusieurs DomainServices peuvent partager la même entité
- Génération de code avec T4: le proxy client peut être généré grâce à notre propre template T4
- Et quelques autres nouveautés comme le support de Windows Azure Storage, amélioration du DataForm...

## ASP.NET MVC3 : syntaxe Razor et nouveautés

Je ne m'attarderai pas sur les dernières versions MVC 3 RC2 mais plutôt sur MVC 3 en général. Rappelons que cette version est compatible avec la précédente. ASP.NET MVC3 apporte pas mal de nouveautés par rapport à la version précédente en commençant par une nouvelle syntaxe qui vient s'ajouter à celle que nous connaissions d'habitude: RAZOR. Cette syntaxe est plus simple et pratique à utiliser. A la création d'un projet ASP.NET MVC 3, nous sommes invités à choisir le type de syntaxe que nous souhaitons utiliser lors de la conception de nos pages. Ces nouvelles pages n'ont plus l'extension .ASPX mais .CSHTML et utilisent la lettre '@'. A noter que peu importe le type de projet choisi, nous pouvons faire cohabiter fichier CSHTML et ASPX. A chaque création d'une nouvelle fenêtre, il nous sera demandé le type de langage pour la page : [Fig.4]. Un petit exemple d'utilisation de cette syntaxe :

```
1. L'heure du jour @DateTime.Now.ToLongTimeString()!!
2.
3. <!-- Single statement block. -->
4. @{ var jour = DateTime.Now.ToLongTimeString(); }
5.
6. @if(true) {
7. <p>Hello World!</p>
8. }
```

Microsoft propose des "helpers" pour ce langage. Pour cela, direction le menu contextuel de notre projet, "Add Library Package Reference" et nous choisissons microsoft-web-helpers. Ceci ajoute des classes d'aide à l'utilisation de twitter, bing,



graphique sous forme d'image, l'antiforgery, le traitement de données JSON, l'affichage de données dans une WebGrid ou l'envoi d'email grâce à WebMail. L'antiforgery permet d'éviter des attaques type "Cross site request". Ici pas besoin de se casser la tête, @AntiForgeryToken.GetHtml() crée un champ caché avec un CRC sur notre page et @AntiForgeryToken.Validate() vérifie sur la page suivante si le champ hidden correspond bien. Les controllers ont eux aussi subi un petit relifting. Nous pouvons utiliser les filtres globaux, la nouvelle propriété dynamic ViewData ou encore les nouveaux ActionResult (HttpNotFoundResult, RedirectToRoutePermanent...). Côté modèle, nous avons une amélioration de la validation grâce à l'interface IValidatableObject. Nous pouvons créer une méthode Validate qui énumérera toutes les validations à traiter.

## IIS Express : du neuf dans le serveur web

Ce serveur web fait son apparition dans WebMatrix. Aujourd'hui nous testons nos applications soit avec Cassini (le serveur web embarqué dans VS) soit avec IIS. IIS Express est un bon compromis. Après l'installation, direction le menu contextuel du projet pour sélectionner "Use IIS Express".

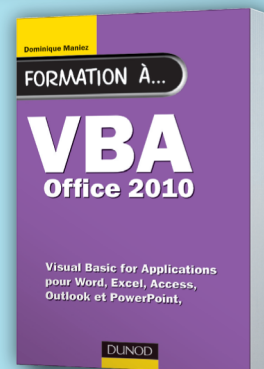
Les avantages sont nombreux comme le support de SSL tout en étant à la fois presque aussi léger que Cassini.

## ADO.NET: EF Code First

Nous connaissons la création d'un fichier edmx à partir d'une base de données. Puis récemment le mode "Model First" pour générer la base de données à partir d'un diagramme de classe. Lors d'une création d'un nouveau projet (ne parlons pas d'un projet avec une base existante) nous avons

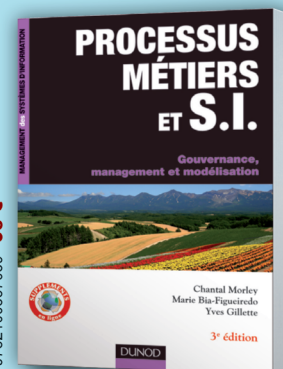
## LES OUVRAGES UTILES POUR VOTRE MÉTIER

Les concepts de base de VBA



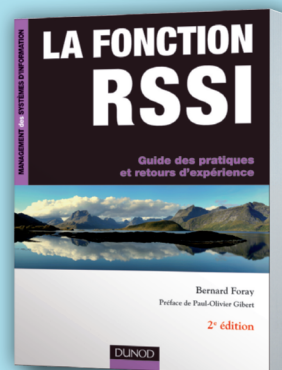
9782100553921 - 19 €

9782100557059 - 38 €



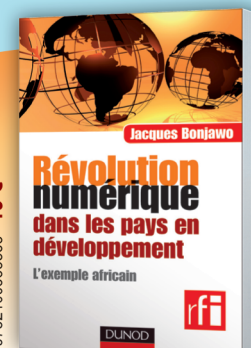
Les repères pour mettre en œuvre le processus métier

Le guide pratique pour sécuriser les données de l'entreprise



9782100556939 - 37 €

9782100553853 - 19 €



Les nouvelles technologies : une opportunité pour les pays en développement

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



Retrouvez-nous du 8 au 10 février sur les TechDays

aujourd'hui trois possibilités (attention à prendre avec humour) :

- Merisien Style (tranche d'âge 30 ans et plus)

Vous êtes attachés à un principe fondamental: pour vous une base de données bien faite c'est un logiciel bien fait. Vous ne faites pas confiance aux assistants. Vous générez d'abord la base de données à la main pour ensuite créer le fichier de mapping O/R.

- Functional Style (20-30ans)

Vous êtes plus UML ou aspects fonctionnels, vous pensez que la base de données n'est qu'un outil de stockage. Vous faites partie de ceux qui pensent que le code type procédure stockée n'a rien à faire dans un SGBD. Le model first est pour vous LA solution pour garder la base de données au service du logiciel... et pas l'inverse ! A noter qu'il faut quand même passer derrière le script généré pour optimiser pas mal de choses (sans compter si vous développez pour SQL Azure)

- Geek Style (le geek n'a pas d'âge, comme Chuck Norris)

Base de données ? Fonctionnel ? C'est quoi ça ! Code Code Code et la base de données suivra ! Le mode Code First est fait pour vous !

Le code first n'est pas que pour les geek puisqu'il nous permet d'avoir un contrôle total sur le code généré. Et j'avoue être assez séduit par cette approche ! Ceci étant dit, voyons comment nous pouvons implémenter une base de données par le code avec un exemple bateau: le fameux Client->Orders. Nous y reviendrons dans un prochain article !

## Visual Studio Async : la killer fonction de C# 5.0 ?

Annoncé lors de la PDC, ce nouveau framework fera certainement partie intégrante de C#5. Il permet de s'abstenir de la création de délégués et méthodes callbacks lors d'appels asynchrones. On a l'impression de faire de la programmation synchrone alors que l'asynchronisme est bien là ! Le téléchargement de la CTP est déjà disponible. Je vous préviens, elle ne fonctionne pas avec le SP1 de Visual Studio pour le moment. Il faut attendre une nouvelle build de la part de Microsoft. VS Async fonctionne en Silverlight mais pas encore pour les appels à WCF. En effet, le proxy généré côté client ne prend pas en compte le type Task<T> (mais devrait l'être prochainement). Nous avons à disposition deux mots clés: async et await et une classe Task<T>.

- Task/Task<T>: représente une action asynchrone
- async: quand une méthode va utiliser de l'asynchronisme avec le mot clé await, il faut spécifier le mot async dans sa signature
- await: c'est le mot clé qui nous évite de créer un callback

Voici un exemple de code qui charge des sites et affiche leurs noms dans une listbox une fois le téléchargement terminé. Dans notre exemple, nous utilisons un projet Silverlight, il ne faut pas oublier d'ajouter une référence vers AsyncCtpLibrary\_Silverlight.dll (My Documents\Microsoft Visual Studio Async CTP\Samples\AsyncCtpLibrary\_Silverlight.dll).

```
1. private void button1_Click(object sender, RoutedEventArgs e)
2. {
3.     LireUrl(new List<Uri>(){ new Uri("http://google.com"),
new Uri("http://dotnet-france.com")});
4. }
5.
6. public void LireUrl(List<Uri> urls)
```

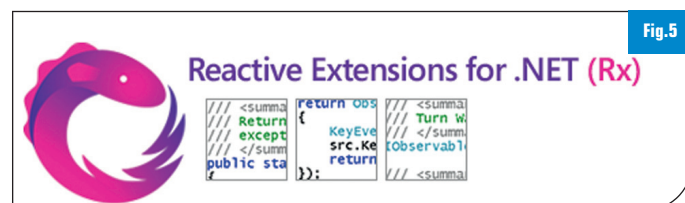
```
7. {
8.     WebClient client = null;
9.     foreach (Uri url in urls)
10.    {
11.        client = new WebClient();
12.        client.BaseAddress = url.ToString();
13.        client.DownloadStringCompleted += delegate(object sender,
DownloadStringCompletedEventArgs e)
14.        {
15.            Dispatcher.BeginInvoke(() => listBox1.Items.Add((sender
as WebClient).BaseAddress));
16.        };
17.        client.DownloadStringAsync(url);
18.    }
19. }
```

Voilà comment nous aurions pu l'écrire avec await/async :

```
1. public async void LireUrl(List<Uri> urls)
2. {
3.     WebClient client = null;
4.     foreach (Uri url in urls)
5.    {
6.        client = new WebClient();
7.        client.BaseAddress = url.ToString();
8.        string content = await client.DownloadStringTaskAsync(url);
9.        Dispatcher.BeginInvoke(() => listBox1.Items.Add((sender
as WebClient).BaseAddress));
10.    }
11. }
```

Il faut réellement se déshabituer d'utiliser les délégués. Ce peut être très déroutant au début mais on y prend vite goût. La méthode DownloadStringTaskAsync est une méthode d'extension qui renvoie une Task<string> au lieu d'un string. Cette méthode est définie dans la référence que nous avons rajoutée à notre projet. Task<T> est le type de résultat que renvoie une méthode utilisée avec le mot clé await. Vous pouvez retrouver encore plus d'exemples sur l'asynchronisme ici : <http://www.wischik.com/lu/AsyncSilverlight/AsyncSamples.html>.

VS Async ne doit pas être confondu avec les RX Extensions (aka linq to events) [Fig.5].



Ce framework est un autre moyen pour jouer avec des processus qui se synchronisent. En attendant un article complet, une phrase que j'ai trouvée sur le net et qui illustre bien la différence entre RX et Async est la suivante : "«Async/await» is not a replacement for Rx any more than LINQ was a replacement for SQL"

Windows Azure subit lui aussi un sérieux lifting : nouveau portail (Silverlight), nouveau rôle (VM Role), création d'un réseau privé (Windows Azure Connect)...

■ Julien Dollon

Responsable Pôle UX - Exakis - MVP Client APP DEV



# Test du **NAS MyDitto** : un ami pour son poste de travail

**C**e nouveau NAS s'appelle « MyDitto » et est fabriqué par Dane-Elec, marque réputée. Le MyDitto se définit comme un appareil de stockage que vous pouvez posséder chez vous et se décompose en 2 parties :

- Un boîtier contenant un ou plusieurs disques durs suivant le modèle (1 ou 2 baies) permettant de bénéficier du RAID.
- De une à plusieurs clés USB suivant les modèles. Le nombre de clés permet d'avoir un responsable administration du boîtier, et les clés supplémentaires, de définir les personnes qui pourront accéder à ces données.

## Fonctionnement

Le principe de fonctionnement est basé sur le principe du peer-to-peer, et le dialogue logiciel se faisant par la clé USB, les données restent en totalité sous votre contrôle. Par ailleurs, un mot de passe vous est demandé à chaque connexion pour être certain d'accéder à votre propre espace privé. Ce principe de clé va vous permettre d'accéder à vos différents fichiers depuis n'importe quel ordinateur dans le monde entier, grâce à son application installée sur cette simple clé USB. La compatibilité de la clé envers le boîtier s'effectue en 2 étapes. La première étape nécessite une configuration de celle-ci avec la mémorisation du numéro de ce matériel, en branchant cette clé sur cet appareil. La deuxième étape qu'effectue l'appareil, concerne la transmission de différentes applications pour une utilisation sur les divers systèmes d'exploitation :

- Windows (XP, Vista, 7)
- MAC OS 10.5 et plus
- Linux (Fedora, Ubuntu, RedHat)
- Téléphone : Windows mobile, iPhone/iPad, Android

Après quelques minutes, vous possédez votre propre NAS avec une double sécurité : la clé contient l'application et un mot de passe en plus pour l'utilisation.

## Utilisation

En branchant la clé USB dans votre ordinateur, vous n'aurez aucun paramétrage à effectuer pour rendre l'application compatible avec votre environnement. Mais une seule opération sera nécessaire : Un double clic pour ouvrir cette clé et exécuter le programme pour bénéficier de différentes utilisations possibles.



Tout d'abord vous vous trouvez devant une interface simplifiée pour communiquer entre le serveur et le disque dur. Par ailleurs, vous pourrez paramétrer une notion de backup provenant de différents ordinateurs vers cet appareil. En plus, vous pourrez synchroniser vos fichiers envoyés depuis les clés MyDitto.

Chaque application possède les mêmes fonctionnalités quel que soit votre système d'exploitation. Tout d'abord, lorsque vous sauvegardez vos documents ou fichiers, certaines fonctionnalités sont disponibles comme la gestion du « Drag and Drop » permettant d'utiliser le copier/coller. Bien sûr, le transfert par navigation est toujours possible, comme cela une file d'attente est automatiquement créée. Du côté de l'accès distant, vous ne passerez pas par le biais de l'internet, qui va freiner vos habitudes de transferts de fichiers en local. Mais vous pourrez lire un document ou visualiser une présentation vidéo sous la forme de streaming. Le produit est actuellement en constante évolution pour laisser entrevoir de nouvelles possibilités pour les années à venir. Une version PRO serait déjà à l'étude pour permettre de couvrir l'ensemble des demandes. <http://www.my-ditto.com/>

### ■ Christophe Villeneuve

Consultant pour Alter Way solutions, auteur du livre « PHP & MySQL-MySQLi-PDO, Construisez votre application », aux Éditions ENI. Rédacteur pour [nexus.net](http://nexus.net), Trésorier AFUP et membre du [LeMug.fr](http://LeMug.fr), PHPTV, PHPteam...

■ **Nvidia** veut marquer son territoire en dévoilant le projet Denver. Cette nouvelle architecture repose sur ARM et doit combiner le CPU et la



GPU afin de disposer d'une unique puce. C'est ce que l'on appelle des processeurs hybrides comme en propose AMD et dans quelques mois Intel. Aucune date de disponibilité n'a été avancée.

■ **AMD** propose une approche similaire avec le concept APU : Accelerated Processing Unit. S'il s'agit toujours d'intégrer dans une même puce CPU et GPU, l'APU propose une architecture différente. APU a pour objectif d'optimiser, d'améliorer la communication et le travail entre les unités CPU et GPU de la puce tout-en-un. Il s'agit de limiter les interfaces de communication et de travailler autour d'une unité unique, l'APU. Pour AMD, il n'y a pas assez d'intégration et trop d'unités intermédiaires. Les principales unités de base du processeur reposent sur une unique couche de transport alors que jusqu'à présent, il en existait plusieurs, brisant les performances.

Ces premières puces concernent les portables et netbook, et AMD espère sans doute détrôner Intel et sa plateforme Atom.

# 1&1 HÉBERGEMENT

J'ai choisi 1&1 car ils bénéficient de serveurs d'une grande stabilité, de coûts abordables, d'une gamme de services très complète et d'une bonne réactivité en hotline.

Sylvain Jacquet  
[www.lemagasinbio.com](http://www.lemagasinbio.com)



# VOTRE SITE TOUT COMPRIS GRATUIT PENDANT UN AN !\*



\* Le Pack Initial est gratuit pendant 1 an sous réserve d'un engagement de 24 mois. Frais de mise en service de 5,97 € TTC. À l'issue de la première année, le Pack Initial est au prix habituel de 2,38 € TTC/mois. Offre domaine applicable uniquement la première année au lieu du prix habituel de 6,99 € HT/an (8,36 € TTC). Conditions détaillées sur [www.1and1.fr](http://www.1and1.fr). Offres sans engagement également disponibles.





# 1&1 PACK INITIAL 1 AN GRATUIT\*

**1&1, des packs hébergement tout compris au meilleur prix !**



## Nom de domaine inclus

Profitez d'un nom de domaine inclus pendant toute la durée de votre pack, sans coût supplémentaire.



## Trafic illimité

Restez à l'abri des mauvaises surprises sur votre facture grâce au trafic illimité.



## Logiciels avancés de web design

Concevez votre site comme un pro et optimisez-le pour la consultation sur terminaux mobiles.



## Fonctions pour développeurs

Disposez de toutes les fonctions nécessaires pour développer votre site et vos applications dans votre environnement favori.



## Outils de marketing en ligne

Propulsez votre site en tête des moteurs de recherche à l'aide des outils de référencement 1&1.



## Centres de données verts

Réduisez votre impact sur l'environnement en optant pour un hébergement 1&1.

## 1&1 PACK INITIAL :

- Nom de domaine inclus
- 2,5 Go d'espace disque
- Trafic ILLIMITÉ
- 10 comptes email (POP3 et IMAP)
- Accès FTP
- Base de données MySQL
- e-Boutique Start
- Outils de référencement
- Assistance technique par email et téléphone
- Et bien plus encore...

**0€\*** ~~100€~~  
pendant un an  
(3,38 € TTC/mois)

Retrouvez toutes nos offres du moment sur [1and1.fr](http://1and1.fr), comme par exemple le nom de domaine en **.fr à seulement 3,99 € HT/an** (4,77 € TTC/an)\* !

# 1&1

Appelez le **0970 808 911** (appel non surtaxé) ou consultez notre site Web

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

# Choisir sa plateforme de blog !



**F**ini le site personnel des débuts de l'Internet. Nous ne sommes plus en 1995 ou en l'an 2000.

Désormais, on parle Twitter, Facebook, Google, réseau social à gogo. Mais tout n'est pas durable. Ainsi, les espaces MySpace sont en perte de vitesse depuis deux ans : moins de pages, chiffre d'affaires en baisse. Il faut dire que Facebook est désormais le réseau social par excellence, même si d'autres éditeurs veulent s'imposer comme Google, Microsoft, Apple.

Il y a un domaine qui ne cesse de croître et même d'exploser : les plateformes de blog. Aujourd'hui, le blog ressemble plus à un gestionnaire de contenu sur-équipé qu'au blog d'il y a 10 ans. Et nous avons le choix entre plusieurs poids lourds. Les plateformes en ligne comme Blogger ou wordpress.com et les plateformes à déployer (avec deux principales offres : Dotclear et Wordpress). Pour celui qui veut se lancer dans le blog, le choix peut paraître cornélien. Si vous ne voulez pas

perdre du temps à gérer l'hébergeur, la configuration, passez par une plateforme hébergée et accessible en ligne comme Blogger. Pour avoir une maîtrise totale de sa plateforme, réaliser des sites professionnels, mieux vaut passer par le déploiement de la plateforme et là, deux poids lourds s'affrontent (même s'il en existe d'autres, notamment basés sur ASP.Net) : Dotclear et Wordpress. Pour le développeur web, le webmaster, ce choix sera le meilleur.

Dotclear ou Wordpress ? Ces plateformes ont considérablement renforcé leur gestion, leur panoplie fonctionnelle grâce à une architecture modulaire et extensible. Et elles se rapprochent des véritables gestionnaires de contenus (CMS). Entre les deux, vos cœurs balancent ? Ils sont finalement relativement proches, la différence se fera sur des fonctions particulières, la

philosophie de fonctionnement. Pour notre part, nous utilisons les deux. Et nous n'arrivons pas réellement à trancher. Nous préférons Wordpress pour son éditeur complet, son tableau de bord, sa stabilité. Nous aimons Dotclear pour sa rapidité de déploiement bien qu'il soit parfois moins sensible que Wordpress, même si la qualité des plug-ins n'est pas toujours au rendez-vous (idem pour Wordpress d'ailleurs).

Reconnaissons que Wordpress est fort de sa plateforme wordpress.com, de sa communauté et de la qualité de son support et de sa documentation, ce qui n'est pas toujours le cas de Dotclear.

Dans ce mini-dossier, vous allez pouvoir vous faire votre avis, comparer, apprécier. A vous de jouer maintenant !

■ François Tonic



# Blog ou CMS, que choisir ?

Nous vivons une époque qui permet à chacun de s'exprimer sur Internet, donc devant une grande partie du monde, au moyen d'un simple blog. Le seul embarras est dans le choix des outils que l'on va faire et éventuellement de son budget.

**P**our faire le bon choix, il convient de bien définir ce que l'on veut, si ce que l'on compte faire est destiné à évoluer, et aussi tenir compte des objectifs visés en termes de visibilité sur le net, autrement dit de référencement et donc de possibilités d'actions sur la génération des pages publiées, ainsi que de ses compétences en administration système. S'il n'est question que de faire un "petit" blog, pour quelques messages épisodiques, on se tournera vers une plate-forme de blog telle que Blogger.com ou Wordpress.com. Les avantages évidents d'une telle solution sont l'immédiateté et la gratuité. Avec une plate-forme comme Wordpress.com, pour prendre cet exemple, il ne faut qu'une petite minute pour créer son blog, et ceci gratuitement. On peut commencer à publier avec un logiciel excellent, qui va générer des pages avec une très bonne sémantique HTML [Fig.1]. C'est très bien. Aucune administration du serveur n'est à assurer. Mais il y a des inconvénients. D'abord la version du logiciel WordPress est bridée en termes de fonctionnalités. L'espace disque est limité. Si vous souhaitez augmenter cet espace ou, par exemple, diffuser des vidéos, ou tout simplement si vous souhaitez personnaliser vos pages avec une feuille de styles CSS, ce ne sera plus gratuit, et peut-être même rapidement assez cher comparativement à d'autres solutions. Enfin, en tant que lecteur de Programmez!, vous pourrez très rapidement ressentir comme frustrante l'impossibilité d'agir sur le code HTML des pages générées. De même si vous voulez publier du code source bien formaté. L'absence d'une taxonomie riche (catégories) ou d'un nom de domaine sera sans doute rapidement frustrante également.

## Le moyen terme : les moteurs de blog

Nous retrouvons WordPress, ou par exemple DotClear. Ces logiciels sont

dits des moteurs de blog, car sans doute, à leur début, c'est ce qu'ils étaient. Il ne faut pas s'y tromper, désormais ces outils méritent le nom de CMS à part entière. Extensibles, ils peuvent être enrichis de nombreux modules. Ainsi un 'blog' WordPress peut évoluer vers un site de e-commerce, grâce à ces modules. Il en va de même pour DotClear cité plus haut. Ainsi le portail de blogs de France 2 (<http://blog.france2.fr/portail.php>), propulsé par DotClear, n'a pas un look de simple blog, mais bien celui d'un vrai site, comme on peut le voir [Fig.2]. Ces outils nécessitent un serveur, ce qui n'est pas gratuit. Mais les offres de serveurs mutualisés à des tarifs raisonnables abondent. PHP et MySQL y seront présents à coup sûr. Il vous suffira de créer une base de données, de déposer les sources du moteur de blog, et c'est parti ! Ceci avec votre propre nom de domaine. Avec ce genre de solutions, le geek pourra agir sur le code HTML des pages pour améliorer son référencement, ajouter des modules, bidouiller des fonctionnalités... bref, se faire plaisir. L'administration du système reste réduite au minimum, et l'installation très rapide. Mais on rencontrera des limites. D'abord au niveau des ressources matérielles : espace disque, mémoire, temps CPU. Ensuite un serveur mutualisé peut être par moment plutôt lent s'il héberge de nombreux sites à fort trafic. Enfin, toujours en ce qui concerne le trafic, un quota peut vous être attribué, qui s'il est dépassé, entraînera une facturation supplémentaire. Et comme les geeks ne rêvent que de sites à fort trafic, il faut y penser :) En outre, on pensera à vérifier que le serveur mutualisé dispose de l'accès distant ssh ou ftp qui va bien pour le travail d'administration avec l'outil de blog/CMS. Ce moyen terme est sans doute idéal pour un blog au départ, susceptible d'évoluer au fil du temps en site, par exemple pour présenter un projet open source, publier de la docu-

mentation technique, peut-être mettre en place un petit forum, etc.

## CMS et serveur dédié

La grosse artillerie, tout confort et toutes possibilités. Y'a-t-il des limites à ce qu'il est possible de faire avec un CMS couteau suisse, comme Drupal pour ne citer de celui-là sur un serveur dédié ? Ou bien avec un CMS écrit en Java, chose a priori impossible sur un serveur mutualisé ? Le seul inconvénient de cette solution est le prix, même s'il existe des offres de serveurs dédiés à des prix intéressants. Cette approche sera réservée aux sites ambitieux, qui dès le départ seront conçus pour être plus qu'un blog, ou aux geeks. Car un serveur dédié, s'il impose des tâches d'administration système non négligeables, est un jouet formidable :)

■ Frédéric Mazué  
fmazue@programmez.com

*L'interface d'administration d'un blog créé sur wordpress.com.*

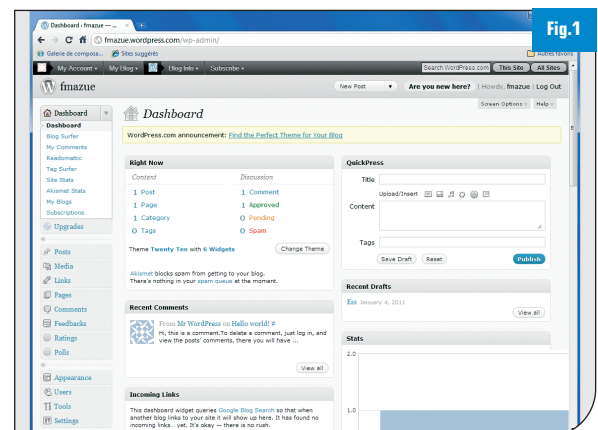


Fig.1



Fig.2

Le site de blog de France2 : plus un vrai site qu'un simple blog.



# Promis, aujourd'hui, je trouve ma plateforme de blog !

Chaque jour, plusieurs centaines de nouveaux blogs font leur apparition dans le monde. Ces blogs proviennent de plateformes dédiées ou de logiciels open source. Mais lorsque vous souhaitez réaliser le vôtre, il n'est pas évident de s'y retrouver. Cet article va tenter de répondre à votre attente.

**U**n blog est une page web contenant des informations personnelles, représentées sous la forme de billets, se triant de façon chronologique, c'est-à-dire le plus récent en première position. Son utilisation peut se définir dans la catégorie des journaux de bord ou des journaux intimes. Le point important étant que le lecteur a la possibilité d'ajouter des commentaires sur un billet précis de son choix, ce qui permet aux blogs d'avoir plus de réactivité et de richesse.

permet pas de vous rémunérer, mais propose l'ensemble des fonctionnalités nécessaires pour réaliser son blog en toute simplicité. Par ailleurs, cette plateforme est allégée aux niveaux des fonctionnalités pour vous permettre d'être réactif rapidement.

<http://www.skyblog.com>

## Overblog

L'objectif de cette plateforme, se veut d'être simple et efficace, sans publicité. Elle est soutenue par une forte communauté et propose de nombreux outils déjà intégrés, évitant en partie d'aller les réclamer. Vous pouvez toujours choisir des options supplémentaires mais payantes.

<http://www.over-blog.com>

## Blog4ever

Cette plateforme est gratuite et propose de nombreuses fonctionnalités très simples. Lorsque vous créez un compte, différentes options vous sont proposées pour réaliser un mini site web avec différentes options : une gestion d'articles, d'albums photos, sondage, forum et newsletter. Ce type de blog est destiné principalement aux blogs "généraliste / actualité". En plus des fonctionnalités de base, vous pourrez ajouter de la vidéo assez facilement. Cette formule permet d'obtenir un mini-site web à petit coût pour un usage personnel, mais de nombreuses options sont disponibles sous licence payante.

<http://www.blog4ever.com/>

## Blogger

Il s'agit de la plateforme soutenue par Google depuis 2003 et qui propose le multi-blog et la possibilité d'être plusieurs à gérer un même contenu. L'offre est gratuite, mais propose de nombreuses fonctionnalités de base

et optimisées comme le référencement et la gestion des flux RSS. Ce type de blog est destiné principalement aux blogs "généraliste / actualité, blog privé / journal intime". En plus des fonctionnalités de base, vous pourrez ajouter de la vidéo assez facilement. <http://www.blogger.com>

## CanalBlog

Il s'agit d'une plateforme clé en main, gratuite et financée principalement par la publicité. Et permet de gérer le contenu d'un même site par plusieurs personnes. Ce type de blog est destiné principalement aux blogs "généraliste / actualité" et accepte le multi-blog. <http://www.canalblog.com>

## Windows Live

Cette plateforme est celle de Microsoft et s'appelait Windows Live Spaces. Depuis quelques semaines, elle se transforme en Windows Live Messenger s'appuyant sur la plateforme Wordpress.com. Aucun changement pour les personnes possédant un compte sur la plateforme, mais de nombreuses évolutions.

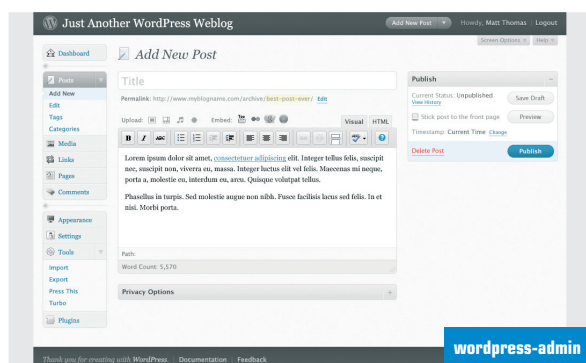
Elle est associée à MSN Messenger et par conséquent gratuite, permettant ainsi de communiquer autour de différents outils associés. Bien sûr, certaines options sont disponibles moyennant paiement.

Ce type de blog est destiné principalement aux blogs "généraliste / actualité, blogs privé / journal intime".

<http://www.windows.live.fr/spaces>

## L'ALTERNATIVE : LES MOTEURS DE BLOGS Wordpress

La version disponible actuellement est la version 3.0.4, distribuée sous licence GNU/GPL. Par ailleurs, une large gamme de plugins est proposée, et



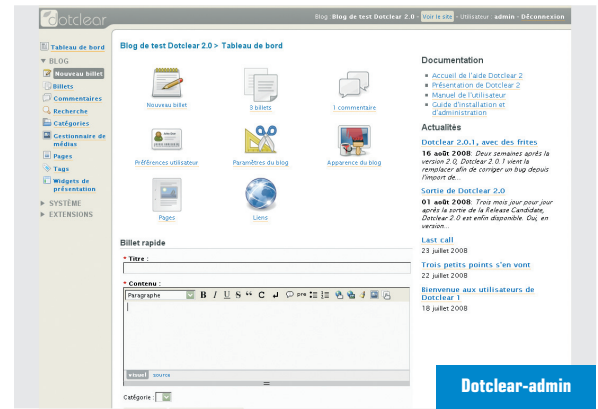


lui permet d'être présent dans la catégorie des CMS. Même si ces évolutions sont disponibles, Wordpress reste principalement utilisé comme une application de blogs par les personnes qui s'en servent. Bien sûr, lors de l'installation, Wordpress peut aussi se configurer comme moteur de blogs pour devenir une plateforme complète. Tout en gardant la possibilité d'utiliser la plateforme mutualisée pour installer votre blog. Cependant, avec le nombre important de plugins, une certaine complexité apparaît et c'est pourquoi, il est souvent utilisé par les internautes avertis où ayant déjà eu plusieurs blogs et qui souhaitent bénéficier d'un maximum de fonctionnalités assez poussées. Bien sûr, l'interface et la gestion restent le plus simple possible permettant de bénéficier de l'ensemble des fonctionnalités disponibles autour d'un blog.

## Dotclear

La version actuelle est la numéro 2.2 et il s'agit d'une réalisation française sous licence GNU/GPL. Depuis la version 2, le multi-blogging permet de proposer directement sur votre site web, une plateforme de blogs pour d'autres personnes. La dernière version a subi de nombreuses évolutions et permet d'être très polyvalent. Elle est très facile d'utilisation pour réaliser son premier blog. Au niveau des fonctionnalités, une large gamme de plugins est disponible, permettant de couvrir l'ensemble de vos besoins pour un site internet classique. Bien sûr, la version de base propose une installation assistée et une gestion administration en multilingue, permettant d'avoir différentes personnes pour administrer son contenu. Une gestion multimédia est présente pour vous permettre d'afficher une galerie photos, ainsi que des documents bureautiques (ODT, PNG...) et un système de Widgets (modules) insérable par le biais du glisser/coller.

<http://fr.dotclear.org>



## CONCLUSION

Nous avons fait un tour d'horizon des différentes possibilités que vous pouvez rencontrer actuellement sur le web. Cependant, le choix de la plateforme est très difficile car à moyen ou long terme, vous pouvez avoir envie de changer, et les conséquences peuvent être assez lourdes comme la perte du référencement ou encore la récupération du contenu existant...

■ Christophe Villeneuve

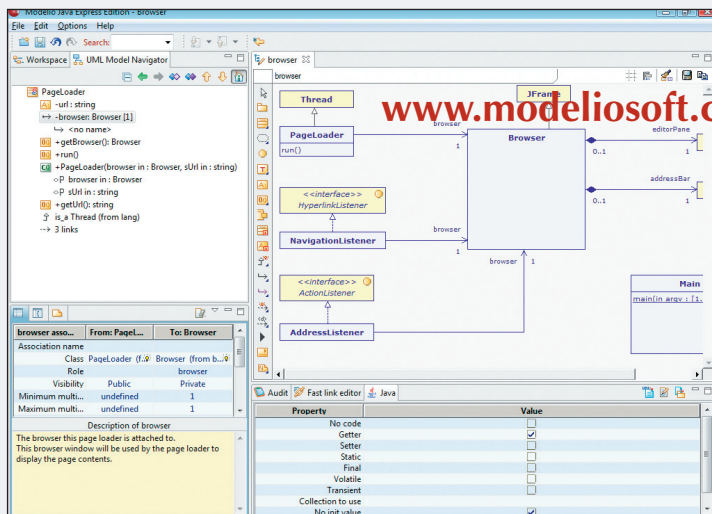


# Modelio : une offre de modélisation unique

*Libérez la vraie puissance de vos modèles !*

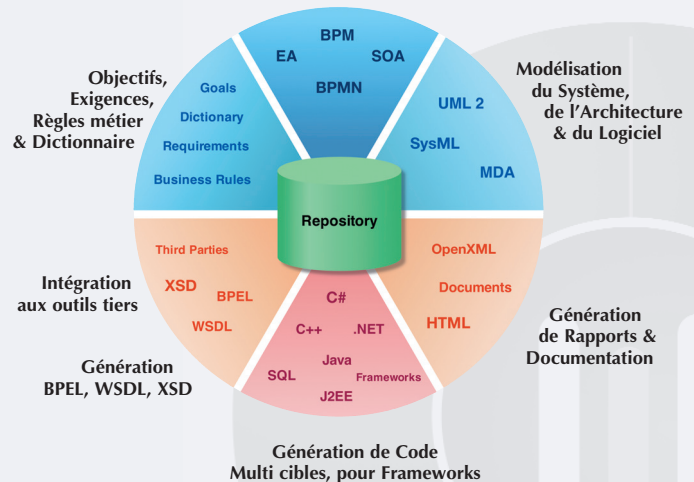
## UML, BPMN, Exigences ..., MDA, génération de code ...

- 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)



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

Modélisation de l'Architecture d'Entreprise,  
des Processus Métier  
& de l'Architecture Orientée Services



## Modelio est disponible en deux éditions

- **Modelio Free Edition** : Un outil de modélisation UML2, BPMN, d'Architecture d'Entreprise et de développement MDA entièrement gratuit !
- **Modelio Enterprise Edition** : La solution de modélisation et de développement MDA complète, supportant le travail en groupe et distribué, extensible avec une riche palette de modules de modélisation et de génération de code disponibles sur étagère



sales@modeliosoft.com - Tél. : 01 30 12 18 40

**Téléchargez et essayez Modelio Enterprise Edition  
gratuitement pendant 10 jours !**

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

# Wordpress, le CMS sur mesure

Wordpress est connu pour être une plateforme de blogging, mais c'est également un CMS particulièrement modulable : fonctionnement en plateforme multisite/monosite, base MySQL, PHP, extensions, API de développement, édition, Web 2.0...

**Q**ue l'on soit à la recherche d'un CMS ("Content Management System" ou Gestionnaire de contenus) pour réaliser rapidement et qualitativement un site Web ou pour exploiter au mieux les ressources du Web 2.0 et des réseaux sociaux - ou les deux à la fois - le choix de Wordpress s'impose comme une option, notamment face à Drupal, son principal "concurrent"

## L'apparence du site

Il y a de multiples façons de traiter l'apparence d'un site WordPress. Les "thèmes" réalisés en conformité avec la version 3.0 ou ultérieure de WP vont pouvoir comporter des options particulières facilement paramétrables dans la console d'administration et une parfaite compatibilité avec les extensions compatibles "Widgets". Les thèmes plus anciens mais pouvant être très intéressants pourront être assez rapidement adaptés par la simplicité d'ajout/modification de balises dans les fichiers de composition de l'apparence du site (header.php, footer.php, page.php...); ces fichiers pouvant être directement édités dans la console. L'ajout de publicité de type Google Ads ou autre ne pose aucun problème.

dans le domaine des sites dits "à contenu". Pas de polémique soulevée, l'idée de cet article est de dévoiler les atouts et la simplicité de mise en oeuvre de ce CMS (même si Wordpress est avant tout une plate-forme de blog, ndr).

## Extensions à l'appel

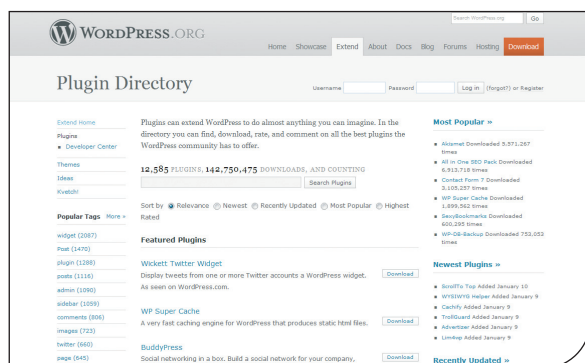
Impossible de lister les extensions disponibles parmi les 12 600 environ dénombrées par Wordpress.org; il y en a déjà au moins trente rien que sur des usages possibles de Tweeter et autant pour Facebook. Ce que l'on peut dire, c'est qu'il est possible depuis déjà plus de trois ans de s'enregistrer un site de podcast auprès d'iTunes, qu'il y a des extensions pour Google Analytics, pour le commerce électronique genre Paypal, qu'il existe des thèmes à foison, etc. L'installation d'une extension peut se faire de deux manières : soit en envoyant des éléments en FTP, soit directement à partir de la section de gestion des extensions de la console d'administration dans laquelle il est possible de rechercher des extensions par mot clé et sélectionner dans la liste qui s'affiche celle que l'on veut installer. Depuis la version 3.0 de WP, les extensions défectueuses sont automatiquement désactivées. La consultation et la recherche d'extensions de WordPress se trouve également accessible sur le site Wordpress.org.

## Interaction extensions/apparence

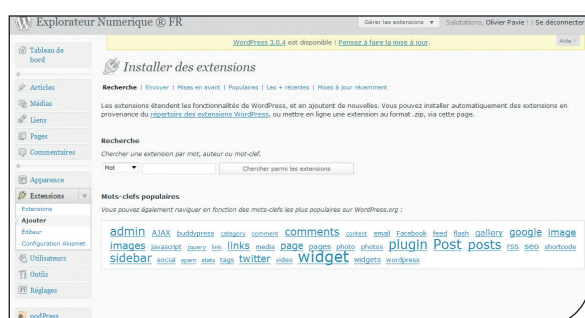
Depuis la version 2.5 de Wordpress, et surtout depuis la version 3.0, l'architecture et la modularité des extensions a évolué pour parfaitement s'architecturer sur le plan logique avec les "themes" qui ne sont rien d'autre que des templates de présentation du site Web en lui-même. De ce fait, les extensions à vocation visuelle telles que nuages de mots, affichage de Tweets ou autres se paramètrent souvent en tant que Widgets et se positionnent donc ainsi facilement dans l'arborescence du site et son apparence; ceci n'empêchant pas non plus un "remodelage" manuel puisque tout cela n'est au final qu'un jeu de scripts PHP, de CSS et de balises HTML. Il faut comprendre en tous cas que ce fonctionnement n'est que la partie émergée de l'iceberg. Le CMS offre une console d'administration, riche et évolutive en fonction des extensions ajoutées.

## Administration visuelle

Entrer dans la console d'administration du CMS est un plaisir visuel. La console affiche de prime abord son Tableau de bord chargé de synthétiser les informations. Informations en provenance de certaines extensions, comme par exemple celles qui servent à comptabiliser les visites, mais également à diffuser de l'actualité en provenance de la



près de 12 600 plugin disponibles



La console d'administration permet de rechercher puis d'installer directement des plugin

## Flux XML en standard

La richesse de WordPress tient dans la génération "en standard" de Flux RSS, ATOM et de divers autres formats à partir des articles qui sont créés. Le nombre d'articles à inclure dans les flux est paramétrable, des extensions ajoutent de nouveaux types de flux pour s'adapter à telle ou telle contrainte. C'est par exemple le cas de l'extension podPress faite pour se conformer aux spécifications de la plateforme iTunes d'Apple et ainsi pouvoir se faire référencer afin de publier un podcast spécifique. Les flux étant potentiellement multiples, leur référencement peut se faire manuellement ou automatiquement (avec certaines extensions). PodPress est par exemple capable de référencement automatisé et multiple d'un flux auprès de plusieurs plateformes avec différents formats. En natif, le flux RSS d'un site WordPress s'extrait par une URL de type : [www.mondomaine.com/feed?=-ATOM](http://www.mondomaine.com/feed?=-ATOM) ou [www.mondomaine.com/feed?=-RSS](http://www.mondomaine.com/feed?=-RSS) ...



communauté Wordpress au sujet de nouvelles versions, de nouveaux Plugin, les commentaires publiés, les articles en mode brouillon, etc. La console d'administration se compose d'une section liée à l'édition du contenu, une seconde à tout ce qui concerne l'apparence, les extensions, les paramètres du site, les utilisateurs et les options des extensions. Pour certaines extensions, une troisième section peut s'afficher en fonction des types de données qu'elles requièrent. Dans l'ensemble, l'accès est toujours bien pensé pour s'y retrouver dans l'administration et le paramétrage de telle ou telle fonction du site. Le mode d'accès au paramétrage des extensions est le plus souvent visuel et synthétique: les plus programmeurs pourront aller travailler sur les "sources" pour affiner l'extension dans un éditeur de texte, soit directement dans la console si l'auteur de l'extension l'a prévu, soit en modifiant le(s) fichier(s) manuellement.

## L'édition pour le Web 2.0

Qu'est-ce qui distingue un site Web 2.0 d'un site Web classique ? La réponse tient dans la gestion du RSS (et autres XML) en standard comme moyen de diffusion des flux d'articles, la possibilité de référencer des mots clé, de gérer des nuages de mots, de supporter les commentaires (ou pas), de traiter les rétroliens et les liens, etc. On ajoute à cela que le CMS intègre des possibilités d'édition des articles et des pages en mode Visuel ou HTML (bien utile pour corriger parfois certains comportements du résultat de l'édition). L'éditeur, appelé TinyMCE, aide à facilement mettre en

forme les textes même s'il manque encore quelques éléments relatifs aux types de polices ou tableaux dans le mode visuel. Toutefois, les extensions pour l'éditeur existent aussi, y compris pour changer les polices ! Pour certains types de caractères par exemple en physique/mathématiques, une extension en AJAX offre la possibilité d'intégrer de vraies formules dans les articles et pages. Pour ce qui est plus "multimedia", il existe des standards intégrés dans l'éditeur comme par exemple l'insertion de photos qui offre une interface très pratique avec un mode de gestion "en galerie" en mode PopUp permettant d'insérer de nouvelles photos mais aussi de ré-exploiter facilement dans un article une photo qui a déjà été utilisée sur le site. La même fonction est utilisable pour des vidéos, des musiques et d'autres médias; la contrainte est la limitation à 64Mo par fichier. Pas de soucis néanmoins puisque pour tout ce qui pourrait être "exotique", l'insertion de balises HTML et de codes dans le mode HTML de l'éditeur permet d'intégrer tout ce que l'on veut, que ce soit des vidéos de toutes provenances, y compris de Youtube, des animations en Flash...

## mode "multisite/monosite"

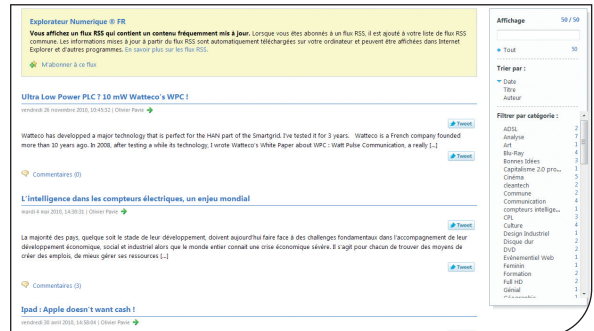
Un des forts potentiels de WordPress se choisit lors de son installation initiale: le CMS peut être déclaré soit en mode de gestion "monosite" (par défaut), soit en mode "multisite". Un simple paramètre déclaré dans le fichier de configuration initiale suffit (voir encadré). Le mode multisite correspond à la version de WordPress

## Une installation très simple

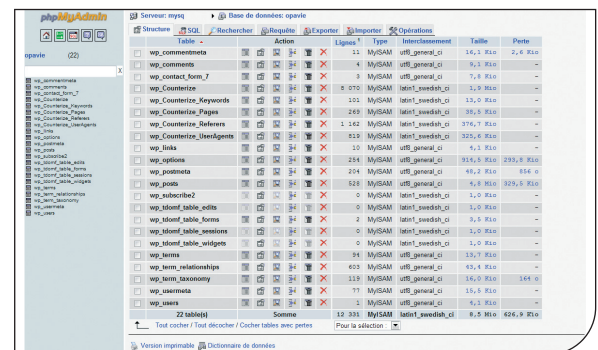
Si certains hébergeurs comme OVH proposent éventuellement la mise à disposition de Wordpress sur l'hébergement choisi, il est très simple d'installer le CMS avec un peu de FTP et un accès à la création d'une base de données MySQL via PHP MyAdmin ou directement depuis une console d'administration de création de bases chez son hébergeur. Une base de 25 Mo peut suffire à gérer n'importe quel petit site. Une fois le CMS chargé en FTP dans le dossier choisi, le fichier de configuration wp-config.php est à paramétrer pour que l'installation sache aller taper dans la base de données pour y inscrire les champs et données nécessaires à la mise en oeuvre basique: l'url d'installation est alors du type [www.truc.com/dossierWP/wp-admin/wp-install.php](http://www.truc.com/dossierWP/wp-admin/wp-install.php). C'est dans ce fichier, avant de lancer la procédure d'installation que se choisit le mode de gestion Multisite/Monosite. Si le fichier de configuration à paramétrer est correct, tout se passe jusqu'à ce que vous arriviez au login et mot de passe à entrer pour arriver dans la console d'administration. Le mot de passe de première connexion est celui qui vous aura été envoyé par mail.

## Planete Wordpress

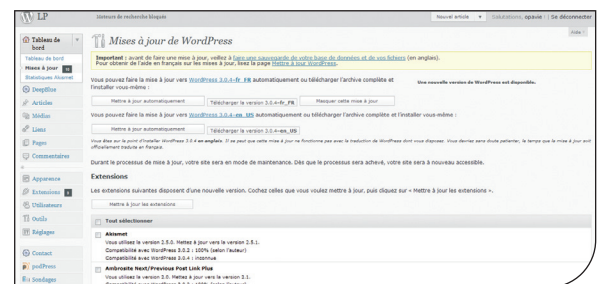
Wordpress, c'est une communauté de développeurs et une communauté d'utilisateurs. Sur Wordpress.com, n'importe qui peut s'ouvrir son blog pour poster son contenu quotidien: c'est le monde du blog qui se trouve en Français sur [fr.wordpress.com](http://fr.wordpress.com). Sur Wordpress.org, c'est la communauté de ceux qui utilisent le CMS pour bâtir un site qui se retrouvent, qu'il s'agisse de développeurs avancés ou d'amateurs éclairés désireux de se créer un site Web "intelligent" et surtout ouvert. [fr.wordpress.org](http://fr.wordpress.org) est la version française officielle de [Wordpress.org](http://Wordpress.org) et [www.wordpress-fr.net](http://www.wordpress-fr.net) est le site officiel de la communauté francophone.



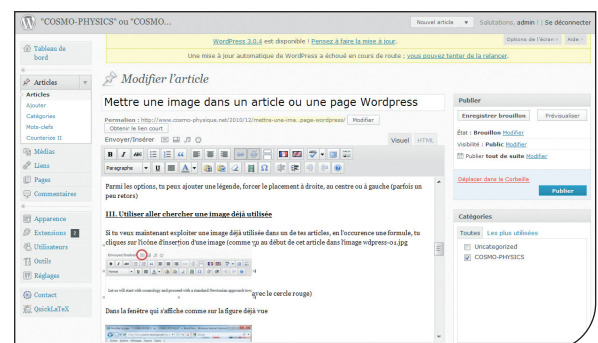
Le flux RSS d'un site tel qu'il est extrait via l'icône RSS du navigateur IEB



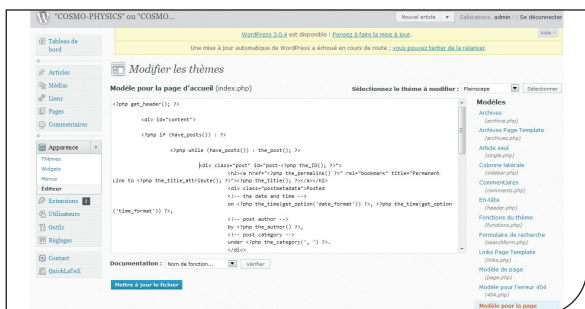
Exemple de petite base MySQL WordPress sous phpMyAdmin



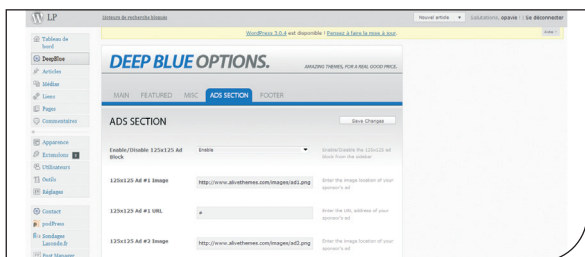
Le tableau de bord récapitule les mises à jour à envisager de faire



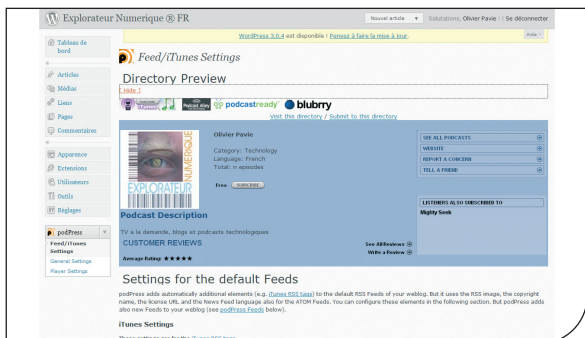
Dans le mode Visuel, l'éditeur de texte permet de travailler en WYSIWYG



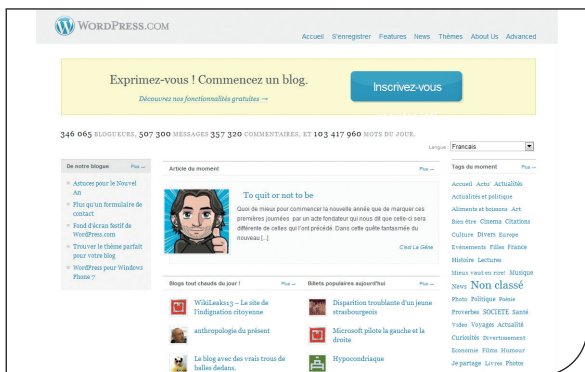
L'apparence du site Web peut être affinée en éditant directement les fichiers d'en-tête, de page, de CSS,...



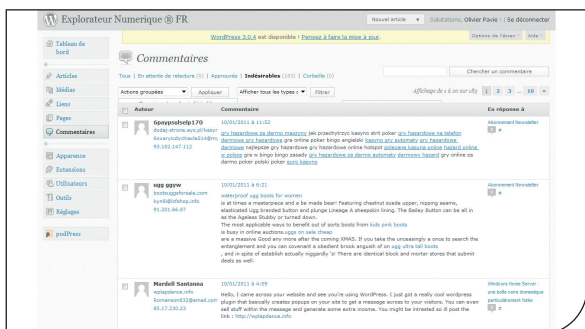
Les options de paramétrage de ce thème Deep Blue vont jusqu'au paramétrage de la publicité à insérer.



Les paramétrages du plugin podPress assurent une compatibilité optimale avec la publication de podcasts sur iTunes.



ne pas confondre WordPress.com pour les bloggeurs et Wordpress.org pour ceux qui construisent des sites Web.



début de la liste de spams identifiés par Akismet

qui s'appelait à l'origine WordPress MU qui a été fusionnée avec WordPress il y a déjà un certain temps. En mode "multisite", la console d'administration s'agrément de un bandeau incluant la possibilité de créer des "sous-sites" qui seront gérés par exemple de la manière soussite.nomdedomaine.com. Cette section offre alors pour chaque sous-site la même console "générale" que lors de l'installation en mode "monosite". Il est à noter que l'installation en tant que multisite requiert des droits avancés d'administration des répertoires par le Webmestre de façon à ce que les alias "soussite.mondomaine.com" puissent être directement gérés par le CMS.

## Protections avancées

Le principe d'un Web 2.0 étant l'ouverture via les commentaires, les retrolis, les potentiels de hacking et de spam sont relativement importants. Par défaut, le CMS propose l'usage d'Akismet contre les spams et commentaires de toutes sortes. Akismet (akismet.com) est dédié à cet usage, l'abonnement en est gratuit pour un usage personnel, un seul compte peut être utilisé sur plusieurs sites. Il est important d'activer cette extension pour gagner un temps précieux afin de se focaliser sur le site et son contenu: plusieurs centaines de spams peuvent être automatiquement éliminés par semaine. Pour qu'Akismet soit la plus efficace, il faut régulièrement mettre à jour sa version et celle de WordPress: les deux informations sur les mises à jour sont indiquées lorsque vous entrez dans votre tableau de bord de la console d'administration. La mise à jour de Wordpress peut être automatique par

simple clic dans la console mais il est essentiel de sauvegarder la base MySQL au préalable. Des extensions pour le backup de la base de données existent et sont capables, par exemple, d'envoyer la base par mail. Pour encore baisser les risques de pollution par spams, l'activation de l'approbation des commentaires est absolument indispensable.

## PHP/MySQL et API

Sur le plan de l'apparence du site, la console d'administration offre un accès direct à la gestion des scripts des différentes sections de la structure des pages : en-tête, article, page, pied de page, colonnes. La feuille de style est elle-même éditée très simplement. Pour ce qui est du développement, le site Wordpress.org regorge de documentations pour développer des thèmes, des Plugin, notamment via la Shortcode API, des Widgets aussi bien pour l'apparence que le Tableau de bord... Les détails des plugin sont en général directement éditables à l'intérieur même de la console d'administration et peuvent être modifiés ou adaptés "en live" avec une sauvegarde du script et un rafraîchissement de la page sur laquelle l'extension est utilisée. Mieux vaut travailler avec quelques backup à échanger en FTP et sauvegarder sa base MySQL avant de trop grands changements.

## Conclusion

Wordpress est loin de se satisfaire de la connotation souvent limitative de CMS de blogging qui lui est donnée. C'est un CMS d'édition et de diffusion de contenus multimédia à vocation d'excellent référencement Web 2.0.

■ Olivier Pavie

## Un référencement qualitatif

La multiplicité des possibilités d'interagir sur la forme des flux XML, la documentation des articles et pages au travers de catégories et de mots clé, ainsi que la déclaration automatisable de nouveaux articles auprès de la plupart des moteurs d'indexation de blogs offre à WordPress une position privilégiée dans l'optimisation du référencement sur Internet et notamment sur Google. Evidemment, l'usage d'extensions spécialisées comme le "All in One SEO Pack", les extensions pour partage sur réseaux sociaux améliorent ou facilitent encore le travail. L'usage de WordPress en mode multisite offre une façon efficace et rapide de disposer d'une plateforme de création de multiples flux qu'il sera ensuite facile d'agréger d'une manière ou d'une autre. En bref, bien utilisé, bien configuré, bien paramétré et avec des informations intelligemment documentées, WWP s'avère très puissant dans le domaine du Buzz sur Internet.



**Spread for Windows Forms** à partir de € 739

GrapeCity PowerTools

Feuille de calcul pour les applications Windows Forms, compatible avec Microsoft Excel.

- Accélérez le développement avec les concepteurs de feuilles de calcul, l'Assistant de prise en main et les concepteurs de graphiques
- Renseignement automatique : anticipation de la frappe dans la cellule
- Nouveau - outil intégré de création de diagrammes avec 85 styles
- Nouveau - préserve les .XLS et restaure les fonctions non supportées
- Inclut des apparences prédéfinies ainsi que la possibilité de créer des apparences personnalisées

**Codejock Xtreme Chart Pro ActiveX** à partir de € 175

CODEJOCK

Incluez des diagrammes dans vos applications ActiveX en quelques lignes de code.

- Affiche un ensemble riche de classes de personnalisation et d'amélioration
- Histogrammes, barres de dispersion, barres empilée, 100% barres empilées et barres horizontales
- Graphiques en secteurs 2D/3D, anneaux 2D/3D et tores 3D, ainsi que les secteurs éclatés
- Inclut les lignes standard, à dispersion, à traçage rapide, en escalier et spline
- Inclut aussi les diagrammes de points, de zones, entonnoir, financiers et de Gantt

**FusionCharts** à partir de € 147InfoSoft Global  
empowering human thoughts

Graphiques Flash &amp; JavaScript (HTML5) interactifs pour les applications Web.

- Animez vos applications Web avec des graphiques interactifs et pilotés par les données
- Créez des graphiques AJAX avec des possibilités d'exploration en quelques minutes
- Exportez les graphiques au PDF et les données en CSV directement depuis les graphiques
- Créez des jauges, des tableaux de bord, des graphiques financiers et plus de 550 types de carte
- Adopté par plus de 17 000 clients et 330 000 utilisateurs dans 110 pays

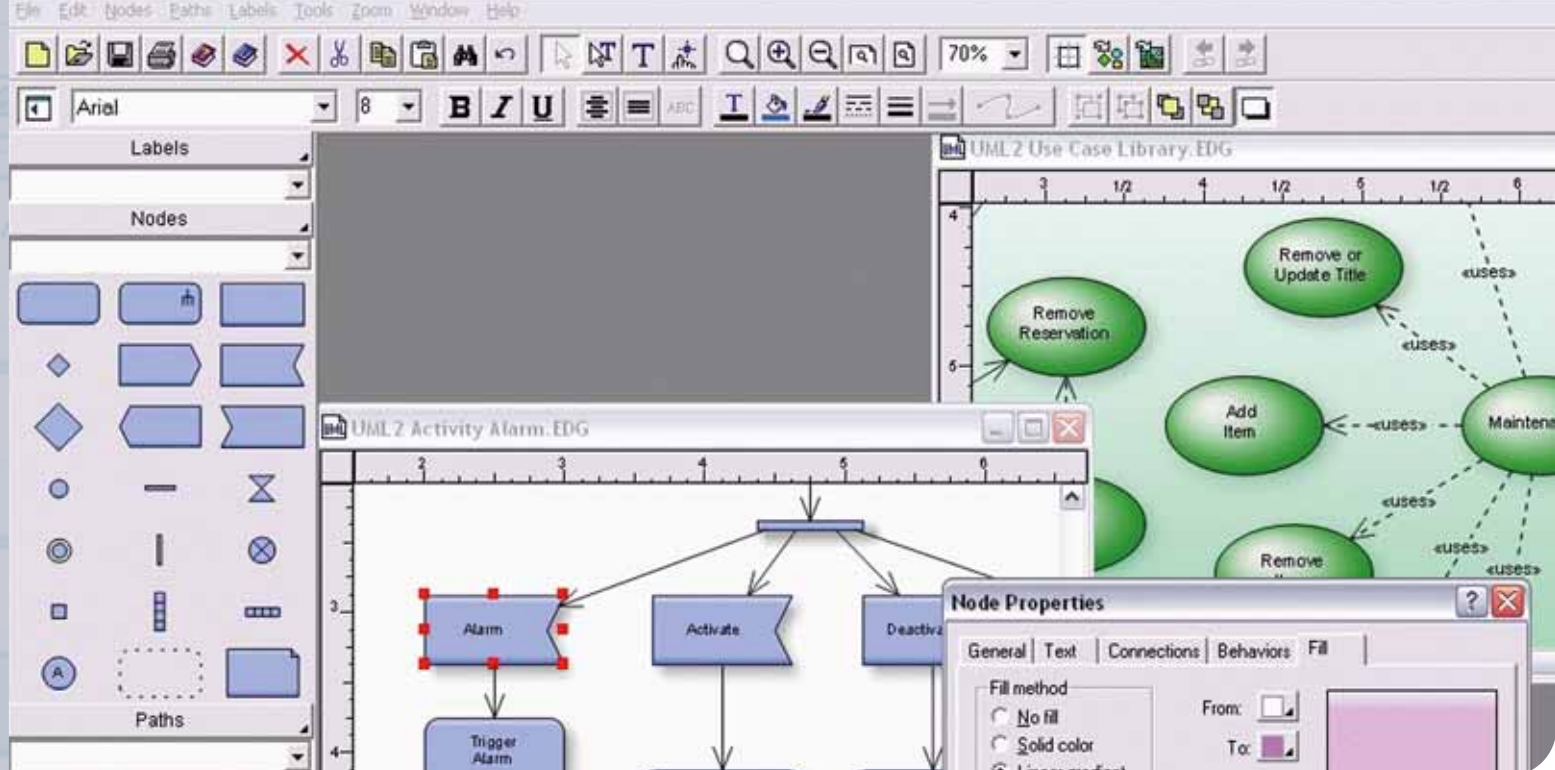
**IBM ILOG Elixir Standard** à partir de € 813

IBM

Étendez les plateformes Adobe Flex et AIR avec des contrôles de visualisation interactifs.

- 10 contrôles d'UI : graphique 3D, jauge, carte, carte de clics, calendrier, OLAP et tableau croisé dynamique, organigramme hiérarchique, arborescence, graphique en radar et chronologie
- Intégration transparente avec l'environnement de développement intégré d'Adobe Flash Builder 4
- Controls can be used individually or in combination, alongside other Flex components
- Utilisation individuelle ou combinée des contrôles avec les autres composants Flex





# Les bases NoSQL vont-elles tuer les bases relationnelles ?

**D**epuis l'avènement du cloud computing, on redécouvre un principe simple : le NoSQL. Pris littéralement, on pourrait croire que NoSQL signifie « pas de SQL ». En réalité, c'est la contraction de « Not Only SQL », pas seulement / uniquement SQL. Ce qui modifie un peu la perception. En tout cas il y a une certitude : NoSQL s'oppose à la notion de données relationnelles, de moteurs relationnels, bref aux SGBDR. Est-ce une notion nouvelle ? Pas réellement, car les SGBD00 étaient souvent non relationnelles. Et on utilise toujours des fichiers plats pour traiter certaines données. Dans l'idée « pas seulement SQL », des langages de type Linq permettent de se passer purement et simplement du langage SQL pour interroger, manipuler les données stockées.

Pourquoi NoSQL revient-il en force ? Plusieurs phénomènes expliquent cette situation : le cloud computing, la volumétrie toujours plus importante des données, la nécessité d'un stockage et d'une interrogation plus rapide et avec une montée en charge en adéquation. Une base NoSQL répond fondamentalement à deux besoins énoncés : montée en charge, stockage des types de données volumineux, et ceci, sans contrainte de structure ni de place. L'explosion des données géospatiales est un exemple éloquent. Oui, les SGBDR savent stocker et utiliser ces données mais est-ce le meilleur endroit pour le faire ? Et les principaux acteurs du cloud computing, proposent des bases de données, des stockages non relationnels, non SQL : Windows Azure Storage (Microsoft), BigTable (Google), Dynamo (Amazon), pour ne citer qu'eux.

Dans un contexte cloud computing, le NoSQL prend tout son sens. Avec les besoins toujours plus considérables de stockage, il faut un SGBD souple et capable de supporter 50 milliards de requêtes journalières (eBay). Mais, le SGBDR classique a sa place sur le cloud, au moins pour migrer, porter les applications desktop et serveur sur le serveur et préserver le modèle de données. Et on peut même observer que finalement, le relationnel et le NoSQL sont finalement complémentaires car ils n'adressent pas les mêmes besoins.

NoSQL est souvent présenté ainsi : fait pour les très grosses bases de données, taillé pour la haute performance et à la très haute disponibilité, accès rapide à la donnée, différentes approches NoSQL, idéal pour le parallélisme et le travail du développeur.

Dans ce dossier, nous allons voir comment et pourquoi utiliser une base NoSQL. De nombreux exemples permettront de mieux comprendre les principes fondamentaux du NoSQL et particulièrement, le premier de tous : les 4 types de bases (orientée colonne, basée sur la théorie des graphes, orientée clé-valeur et orientée document).

Et pour répondre à la question posée dans le titre, non le NoSQL ne va pas tuer le relationnel et le SQL. Il est au contraire complémentaire et le remplace dans certains contextes précis. Le SGBDR a encore de nombreuses années devant lui, qu'on le veuille... ou non.

■ François Tonic

# Vers la fin du tout relationnel !

Cœur des applications informatiques, le stockage des données est la chasse gardée des SGBDR qui profitent de la mainmise du monde relationnel sur la modélisation des données depuis plus de 40 ans maintenant. Et jusqu'alors, aucun concurrent sérieux n'a réussi à bousculer durablement les SGBDR. Cependant, depuis bientôt 2 ans un mouvement alternatif ne cesse de prendre de l'ampleur au point de s'imposer aujourd'hui comme une réelle alternative aux SGBDR et au monde relationnel. Répondant au nom de NoSQL, ce mouvement est présenté en détail dans cet article.

**L**a gestion des données est la chasse gardée du monde relationnel depuis de très nombreuses années maintenant. Difficile dans ces conditions pour des technologies émergentes de venir concurrencer de façon sérieuse le monde des SGBDR. En effet, on ne remplace pas comme cela une plateforme et un savoir-faire améliorés et pérennisés depuis autant d'années et qui surtout ont largement fait leurs preuves ! Néanmoins, l'avènement de grands du Web, comme Google ou Amazon pour ne citer qu'eux, a mis en évidence que les sites à fort trafic s'appuyant sur des bases de données relationnelles ne peuvent garantir un niveau de performance suffisant bien souvent. En effet, la consistance des données garantie par le modèle relationnel coûte cher en performances.

Partant de ce constat, ces grands du Web ont développé sur mesure des solutions innovantes pour répondre à leurs besoins pharaoniques en termes de charge et de volumétrie de données. Ces grands n'ayant pas forcément les mêmes besoins, puisqu'opérant dans des domaines différents (moteur de recherche pour Google et site de commerce pour Amazon par exemple), les différentes solutions qui ont émergé n'ont pas les mêmes caractéristiques et leurs cas d'utilisation peuvent de fait être différents. Ces diverses solutions ont été mises au grand jour durant l'année 2009 lors d'une conférence à San Francisco, fief de ces grands du Web. Le mouvement NoSQL venait de naître officiellement !

Ainsi, NoSQL signifie "Not Only SQL" et regroupe l'ensemble de ces bases de données innovantes ayant pour caractéristiques principales de s'affranchir du modèle relationnel et d'être particulièrement performantes dans des contextes de haute disponibilité. Celles-ci n'offrent donc pas d'interface de requêtes en SQL.

## Un changement de paradigme

Pour mieux comprendre les caractéristiques principales des bases de données NoSQL, il est important de revenir aux racines du monde relationnel. Pour définir la manière dont seront modélisées et stockées les données, on utilise basiquement des entités et des relations. Ces dernières permettant de lier entre elles différentes entités. Le plan contenant l'ensemble de ces éléments est exprimé ensuite sous la forme d'un schéma de bases de données relationnelles. Afin d'éviter autant que possible la redondance d'informations, on cherche ensuite à normaliser cette base de données ainsi définie.

Les opérations successives effectuées sur ces bases de données sont encapsulées au sein de transactions de données. La fiabilité

de celles-ci est garantie par le respect de 4 propriétés fondamentales, facilement mémorisables via les 4 lettres de l'acronyme mnémotechnique ACID dont voici le détail :

- **Atomicité** : l'ensemble des opérations d'une transaction doit être validé ou alors elles doivent toutes être annulées pour revenir à l'état initial.
- **Cohérence** : les modifications apportées à la base de données au cours de la transaction doivent la laisser dans un état consistant en accord avec l'ensemble des contraintes d'intégrité de la base.
- **Isolation** : les transactions démarrées simultanément ne doivent jamais interférer entre elles ce qui signifie que les opérations d'une transaction ne peuvent voir l'état intermédiaire des données modifiées par une autre transaction tant que celle-ci n'est pas terminée. Ainsi, les transactions ne peuvent s'exécuter de manière concurrentielle.
- **Durabilité** : une fois qu'un client a été averti qu'une transaction s'est terminée correctement, l'ensemble des modifications résultant est appliqué et ne sera pas annulé.

Ces propriétés propres aux bases de données relationnelles amènent un certain nombre de contraintes fortes en termes de performances. En effet, il est notamment impossible pour des transactions de s'exécuter en parallèle. Cela peut bien entendu être pertinent dans certains cas mais dans d'autres cette contrainte pénalisante peut ne pas être nécessaire. Cela est bien souvent le cas justement pour les sites à fort trafic des grands du Web. Les bases de données NoSQL qu'ils utilisent s'appuient quant à elles sur le théorème du CAP d'Eric Brewer qui est plus adapté aux données massivement partagées. Les 3 grandes principes de ce théorème sont les suivants :

- **Consistance (C)** : chaque client voit la même vue lorsqu'il y a des mises à jour, ce qui correspond à l'Atomicité des propriétés ACID.
- **Haute Disponibilité (A)** : il est toujours possible d'accéder aux données en lecture / écriture même en cas de panne.
- **Tolérance au Partitionnement (P)** : le système est tolérant au partitionnement.

En revanche, ce théorème dit également que seuls deux de ces trois principes peuvent être respectés en même temps dans un système distribué. Ainsi, suivant les cas d'utilisation pour lesquels ils ont été conçus les systèmes NoSQL pourront être amenés à privilégier tantôt le couple CA (Consistance / Haute Disponibilité) tantôt le couple AP (Haute Disponibilité / Tolérance au Partitionnement).

nement).

C'est ainsi qu'entre en jeu la notion de cohérence in-fine ("eventual consistency" en anglais). Dans le modèle des bases NoSQL, les données sont lues et écrites sur plusieurs nœuds, ce qui induit forcément des redondances. Il appartient désormais au développeur de définir le degré de fiabilité qu'il souhaite obtenir. Cette customisation du degré de fiabilité, qui est à rapprocher du réglage du niveau d'isolation des bases de données relationnelles, devra être maîtrisée par les développeurs qui apprendront également à jongler avec la redondance.

Bien que l'éventail des solutions de bases de données NoSQL soit assez vaste, un certain nombre de caractéristiques techniques communes émergent donc. On peut ainsi retenir une absence de schémas, un partitionnement horizontal sur un grand nombre de nœuds, une dé-normalisation des données, la réplication automatique et le versionning des données. Globalement, le stockage des données est réalisé au sein d'une table de hachage distribuée. Cette dernière va permettre une distribution des requêtes et des données sur plusieurs nœuds. La répartition entre les nœuds se faisant via un algorithme de hachage qui a la lourde tâche de garantir une répartition la plus égale possible afin de fournir une disponibilité satisfaisante.

## Les grandes familles

Au-delà de ces grandes propriétés communes, les solutions estampillées NoSQL restent néanmoins hétérogènes. De fait, il est de coutume de regrouper ces bases au sein de grandes familles. Celles-ci diffèrent notamment sur la forme de représentation des données qu'elles gèrent. Chacune de ces familles ayant ses spécificités qui la rendent plus adaptée à la manipulation d'un certain type de données et donc utilisables dans des cas d'utilisation bien précis. Ces grandes familles sont au nombre de 4 :

- Famille des bases de données clé - valeur.
- Famille des bases de données orientées colonnes.
- Famille des bases de données orientées document.
- Famille des bases de données orientées graphe.

## BD Clé - Valeur

On commence notre tour d'horizon avec la famille des bases de données de type clé - valeur. La représentation des données adoptée par ces bases NoSQL est la plus simple possible puisque qu'on associe à une clé une valeur [Fig.1]. Cette dernière est un bloc de données (chaîne de caractères, objet sérialisé, ...) dont le contenu est inconnu de la base de données sous-jacente. En pratique, ce type de représentation se révèle très performant puisque les opérations de lecture / écriture

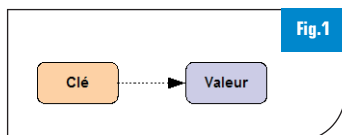


Fig.1

Représentation BD Clé / Valeur

sont réduites à un simple accès disque et qu'il est particulièrement adapté aux solutions de cache. L'API permettant de communiquer avec ce type de base est ultra simpliste puisque composé seulement des opérations suivantes : PUT, GET et DELETE. Enfin, les recherches multi-critères auxquelles nous étions habitués sur les SGBDR sont remplacées par des recherches "full-text", ce qui n'est pas anodin en termes d'architectures applicatives.

Les implémentations les plus populaires de ce modèle sont Redis,

Riak et Voldemort. Dans ce qui suit, nous nous intéressons à la base de données Riak. Ecrit principalement via le langage Erlang, Riak est un système clé-valeur distribué qui offre un niveau de consistance paramétrable. La fameuse notion de cohérence in-fine évoquée précédemment dans cet article ... Cette notion de consistance revient à pouvoir garantir avec un degré de fiabilité plus ou moins important que chaque lecture voit toujours la dernière version de la donnée écrite. Le paramétrage de la consistance s'effectue via la définition d'un quorum à atteindre pour avoir le niveau de fiabilité souhaité et il est défini via les 3 variables suivantes :

- N : le nombre de répliquas pour chaque partition du cluster.
- R : le quorum en lecture, c'est-à-dire le nombre de réponses à attendre lors d'une requête en lecture.
- W : le quorum en écriture, c'est-à-dire le nombre de confirmations d'écriture à attendre lors d'une requête en écriture.

La consistance est parfaite lorsque la condition suivante est vérifiée :  $R + W > N$ . Ainsi, on jouera avec les coefficients afin d'obtenir le meilleur compromis pour les besoins de notre système. Lorsque cette consistance vient à être relâchée, des conflits de version peuvent apparaître et c'est alors à l'application cliente qu'incombe la tâche de résoudre les conflits d'une manière adaptée au métier.

Riak offre une interface REST et possède des API clientes dans de nombreux langages mais pour l'exemple qui suit, nous allons utiliser son interfaçage avec le langage Java afin de réaliser quelques opérations de base :

```
// Connexion à Riak
RiakClient riak = new RiakClient("http://localhost:8098/riak");
// Ecriture d'un objet
RiakObject obj = new RiakObject("bucket", "key", "value");
riak.store(obj);
// Lecture de l'objet
FetchResponse r = riak.fetch("bucket", "key");
if (r.hasObject()) {
    obj = r.getObject();
    // Affichage de l'objet
    System.out.println("Bucket : " + obj.getBucket() + ", Key : " + obj.getKey() + ", Value : " + obj.getValue());
}
```

Cet exemple montre que Riak organise les ensembles logiques clés / valeurs sous forme de buckets. En outre, le paramétrage de la consistance peut être réalisé comme suit :

```
// Définition d'un coefficient de réplication N = 5 pour le bucket
RiakBucketInfo bucketInfo = new RiakBucketInfo();
bucketInfo.setNVal(5);
riak.setBucketSchema("bucket", bucketInfo);
// Ecriture d'un objet en spécifiant le quorum en écriture W = 3
riak.store(obj, RequestMeta.writeParams(3));
// Lecture d'un objet en spécifiant le quorum en lecture R = 2
riak.fetch("bucket", RequestMeta.readParams(2));
```

Bien que le modèle de représentation des bases de données clé / valeur du type de Riak soit simpliste, ces bases constituent une puissante solution de stockage permettant de répartir les don-



# Advanced database technology for breakthrough applications



## Laissez vos applications s'envoler...

Avec InterSystems Caché, faites décoller vos applications. Elles bénéficieront immédiatement de performances hors du commun, deviendront massivement scalable et ne nécessiteront plus d'administration fastidieuse.

InterSystems **Caché**® base de données post-relationnelle, mais aussi serveur d'application, framework Ajax, ... se fonde sur une technologie Objet avancée qui permet de construire beaucoup plus facilement des applications XML, Web Services, AJAX, Java et .NET.

InterSystems Caché est aussi une base SQL jusqu'à 5 fois plus rapide que les bases de données relationnelles classiques en accès SQL et bien plus en accès Objet !

Grâce à son Architecture de Données Unifiée unique, Caché élimine le

besoin de mapping objet-relationnel, réduit les temps de développement, et facilite l'évolution et la maintenance de votre application.

Déployé sur plus de 100.000 systèmes de par le monde pour des applications de 2 à 50.000 utilisateurs, Caché est disponible sur toutes les plates-formes majeures du marché.

**Nouveau:** L'intégration de Caché avec JAVA - JNI permet littéralement d'exploser les performances et d'offrir enfin aux programmeurs JAVA un moteur sans compromis et digne de leurs réalisations.

Depuis plus de 30 ans InterSystems vous apporte des technologies avancées qui vous permettent de construire des applications qui font la différence.

# INTERSYSTEMS

Téléchargez votre version gratuite complète InterSystems Caché - sans limite de temps: [InterSystems.fr/avancee](http://InterSystems.fr/avancee)

nées et la charge entre plusieurs nœuds avec une réplication des données qui augmente la disponibilité de celles-ci.

## BD Colonnes

Les bases de données orientées colonnes ont un mode de représentation des données que l'on oppose traditionnellement à l'orientation en ligne utilisée dans les bases de données relationnelles. En effet, les SGBDR gèrent les colonnes d'une ligne d'une table de manière statique. A contrario, les bases de données orientées colonnes ont une vision bien plus flexible qui permet d'avoir un nombre variable de colonnes pour chaque ligne. Ainsi, chaque ligne peut disposer uniquement des colonnes qu'elle utilise et on évite de se retrouver avec des colonnes ayant des valeurs à NULL pour certaines lignes comme cela peut être le cas dans les bases de données relationnelles.

De fait, il est possible d'avoir un nombre de colonnes très important pour les lignes le nécessitant, sans le coût du stockage des valeurs non utiles qu'on aurait avec une base relationnelle. Enfin, ces bases permettent de regrouper les colonnes d'une ligne en familles de colonnes ce que l'on appelle aussi super-colonnes [Fig.2].

Les bases de données orientées colonnes sont prévues pour stocker des millions de colonnes, ce qui en fait des bases particulièrement adaptées pour le stockage de relations one-to-many.

Les implémentations les plus populaires de ce type de bases sont HBase et Cassandra. Ici, nous nous concentrons sur HBase. Écrit en Java, HBase est un système de base de données orienté colonnes fonctionnant au-dessus du projet Apache Hadoop. Il est ainsi parfaitement intégré avec ce dernier et peut donc s'interfacer naturellement avec une tâche MapReduce d'Hadoop. De plus, HBase est une implémentation Open Source du modèle BigTable publié par Google, une référence en ce qui concerne le traitement de gros volumes de données.

Avec HBase, les familles de colonnes doivent être définies à la création de la base. Ensuite, chaque famille pourra recevoir un nombre arbitraire de colonnes à n'importe quel moment, même après la création de la base. En termes de volumétrie, HBase est prévu pour être performant même en accueillant plusieurs millions de colonnes et plusieurs milliards de lignes. Ceci étant rendu possible en rajoutant des nœuds en conséquence. Le code suivant permet de mettre en avant quelques possibilités de l'API de HBase :

```
HBaseConfiguration conf = new HBaseConfiguration();
// chargement de la configuration HBase où sont définies les
// familles de colonnes (Identité et Divers)
conf.addResource(new Path("/home/sylsau/hbase-conf.xml"));
HTable table = new HTable(conf, "persons");
// Création des lignes à insérer
BatchUpdate person1 = new BatchUpdate("Person1");
person1.put("Identite:Prenom", Bytes.toBytes("Rachida"));
person1.put("Identite:Nom", Bytes.toBytes("Saurel"));
BatchUpdate person2 = new BatchUpdate("Person2");
person2.put("Identite:Prenom", Bytes.toBytes("Sylvain"));
person2.put("Identite:Nom", Bytes.toBytes("Saurel"));
person2.put("Divers:Metier", "Ingenieur");
// Ecriture
table.commit(person1);
table.commit(person2);
// Sélection d'une valeur spécifique
Cell cell = table.get("person1", "Identite:Prenom");
if (cell != null) {
    System.out.println("Prénom : " + Bytes.toString(cell.get
Value()));
}

// Sélection de plusieurs lignes pour une colonne donnée
Scanner scanner = table.getScanner(new String[] {"Identite:
Prenom"});
for(RowResult result : scanner) {
    System.out.println("Ligne : " + Bytes.toString(result.get
Row()) + " avec Prénom : " + result.get(Bytes.toBytes("Identite:
Prenom")));
}
// fermeture du scanner et de la table
scanner.close();
table.close();
```

On voit clairement qu'avec ce type de base de données, on doit repenser entièrement les pratiques habituelles issues du monde relationnel. Ainsi, le modèle de données doit être dé-normalisé afin de respecter l'orientation en colonnes de HBase. La redondance de données qui en résulte n'est cependant en rien pénalisante

avec ce type de base de données puisqu'elles supportent aisément les gros volumes de données. En outre, les valeurs stockées dans les colonnes peuvent être des objets sérialisés ou de simples chaînes de caractères comme dans notre exemple, ce qui confère une grande souplesse à HBase. Les recherches complexes sur les données stockées devront s'effectuer par l'intermédiaire de tâches MapReduce développées spécifiquement.

Les bases de données orientées colonnes offrent une grande souplesse mais restent néanmoins difficiles à appréhender de par leur mode de distribution et leur modèle de données singulier. Ainsi, le développeur devra rester attentif pour éviter certains

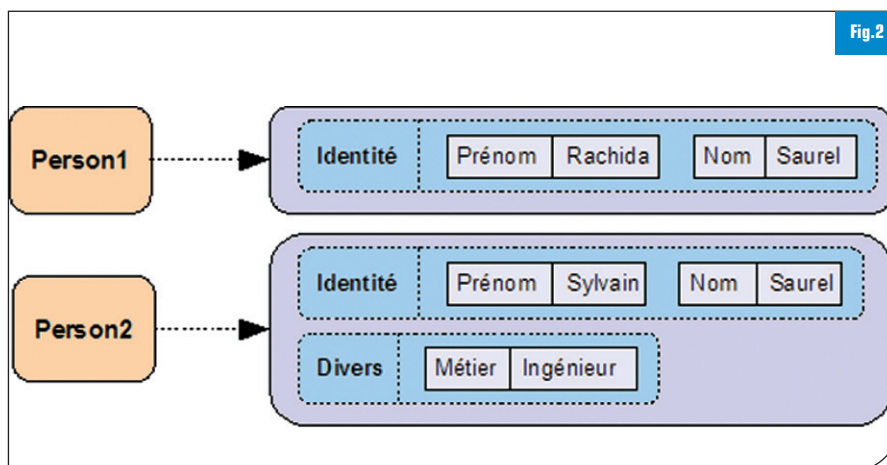


Fig.2

Représentation BD Colonnes

écueils et notamment au niveau des mises à jour qui peuvent s'avérer très coûteuses. En effet, alors que pour une base de données relationnelle, une mise à jour d'un tuple disposant d'une clé étrangère peut suffire, une base orientée colonnes pourra nécessiter une mise à jour de toutes les valeurs d'une colonne pour tous les enregistrements ! Leur puissance leur permet cependant d'être globalement très performantes avec une grande volumétrie de données tout en offrant une haute disponibilité.

## BD Document

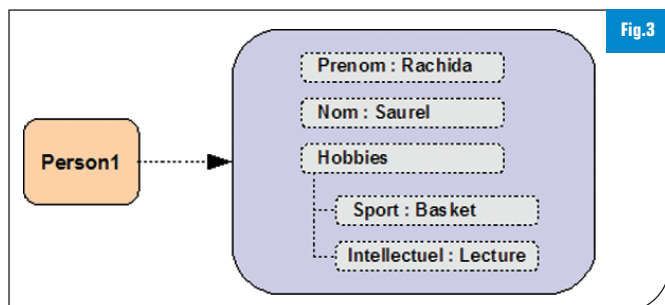
Les bases de données orientées document sont en quelque sorte des bases de données clé - valeur au sein desquelles la base est consciente du contenu de la valeur associée à une clé. On parle de base orientée document car la valeur est un document de type JSON ou XML. Ce dernier possède donc une structure hiérarchique, ce qui permet à ces bases de pouvoir retrouver à partir d'une simple clé tout un ensemble d'informations [Fig.3]. L'équivalent dans le monde relationnel serait coûteux puisqu'il impliquerait un grand nombre de jointures.

La connaissance des informations stockées permet aux bases orientées document d'offrir des fonctionnalités supplémentaires à ce que proposent les bases de données clé / valeur. Ainsi, il sera possible d'effectuer des opérations de type CRUD sur le contenu d'un document. De plus, la structure hiérarchique des documents stockés permet une indexation des champs de ceux-ci, ce qui permet d'interroger la base de données en élaborant des requêtes complexes. Enfin, l'orientation document de ces bases les prédestine au stockage de relations one-to-one et one-to-many, ce qui rend inutiles les traitements de jointure lors des opérations de lecture / écriture sur un document. Le stockage des mêmes informations au sein d'une base de données relationnelle serait bien plus coûteux par la suite lors de ces opérations.

Pour ce modèle de base, les implémentations les plus populaires sont CouchDB et MongoDB. Ici, nous nous intéressons à MongoDB. Ecrite en C++, cette base utilise un format qu'elle a créé : le BSON qui est un dérivé binaire du JSON. L'utilisation de ce format spécifique pour le transport rend obligatoire la présence d'un driver, ce qui peut être considéré comme un inconvénient par certains. L'architecture de MongoDB lui permet d'être extrêmement performante pour des insertions de gros volumes de données grâce à son mode batch.

MongoDB propose une API Java dont nous allons voir quelques exemples de manipulation :

```
Mongo mongo = new Mongo();
// Création de la base de données
DB db = mongo.getDB("database");
```



Représentation BD Document

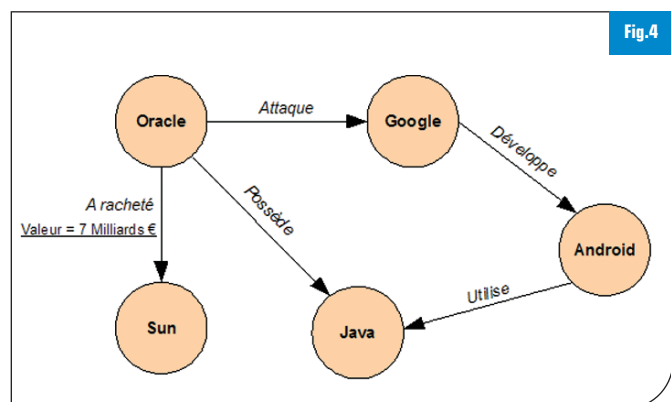
```
// Création de la collection dans laquelle seront stockées les
// personnes
DBCollection persons = db.getCollection("persons");
// Mise en forme des données
BasicDBObject person1 = new BasicDBObject();
person1.put("Prenom", "Rachida");
person1.put("Nom", "Saurel");
Map<String, String> hobbies = new HashMap<String, String>();
hobbies.put("Sport", "Basket");
hobbies.put("Intellectuel", "Lecture");
person1.put("Hobbies", new BasicDBObject(hobbies));
// Ecriture des données
persons.insert(person1);

// Recherche de données avec une requête utilisant un Basic
// DBObject
BasicDBObject query = new BasicDBObject();
query.put("Prenom", "Rachida");
DBCursor cursor = persons.find(query);
// Lecture des résultats
while(cursor.hasNext()) {
    System.out.println(cursor.next());
}
```

Ce rapide exemple montre que les documents sont stockés au sein de collections et qu'un ensemble de collections constitue une base de données. Pour conclure sur Mongo DB, on gardera à l'esprit que cette base de données NoSQL cherche à répondre avant tout à des problématiques de performance en temps réel tout en garantissant une scalabilité horizontale avec des fonctionnalités de partitionnement horizontal et de réplication intégrées.

## BD graphe

Dans l'univers des bases de données NoSQL, les bases de données graphe se démarquent. En effet, là où les solutions des autres grandes familles NoSQL visent à résoudre en priorité des problématiques de performances, les bases de données graphe adressent des problématiques liées à la représentation des données. Leur but premier est donc de pouvoir résoudre des problèmes complexes qu'il serait difficile voire impossible de modéliser avec les bases de données relationnelles. Ces systèmes sont basés sur la théorie des graphes. Ainsi, un graphe est composé de nœuds, de relations et de propriétés de type clé / valeur que l'on peut affecter à ceux-ci. Les données modélisées au sein



Représentation BD graphe



de ce type de bases utilisent donc cette représentation spécifique [Fig.4].

La modélisation des données représentée dans la figure 4 montre très clairement l'avantage de la modélisation via un graphe. En effet, cette dernière est très peu contrainte et on peut représenter de manière fidèle le monde réel. En outre, ces bases de type graphe implémentent un certain nombre d'algorithmes classiques de la théorie des graphes tels que la recherche du plus court chemin ou bien le A\* pour ne citer qu'eux. Ainsi, les cas d'utilisation classiques de ces bases concernent en premier lieu les réseaux sociaux où l'on voit clairement l'intérêt d'une modélisation en graphes. Cela peut également convenir parfaitement pour la modélisation d'arbres généalogiques, de problèmes avec des réseaux de transports, etc.

La base de données graphe la plus mature et la plus connue est Neo4j. Ecrite en Java, elle permet d'être directement embarquée au sein d'une application contrairement aux autres bases de données NoSQL vues jusqu'ici. Bien entendu, elle offre également un mode client / serveur accessible via une API REST. La manipulation d'un graphe est très intuitive car basée sur une API simple et puissante dont le cœur est constitué par les classes Node et Relationship permettant de modéliser respectivement un nœud et une relation.

Neo4j encapsule chaque opération de manipulation de la base au sein d'une transaction. Toutes les opérations réalisées au cours d'une transaction sont persistées automatiquement une fois celle-ci terminée. En outre, on constate que les relations sont typées via des enum Java, ce qui améliore l'efficacité de la validation à la compilation. Ainsi, il est nécessaire de définir un enum implémentant RelationshipType au sein duquel on déclare les types de relations que l'on souhaite manipuler :

```
public enum GraphRelationships implements RelationshipType {
    ACQUIRED,
    USES,
    // etc ...
}
```

Le parcours du graphe s'effectue en se déplaçant de nœuds en nœuds via les relations ou en utilisant l'API Traverser de Neo4j qui donne la possibilité de construire des requêtes de sélection de manière assez intuitive. En termes de scalabilité, Neo4j ne supporte pas pour le moment la distribution d'un graphe sur plusieurs nœuds mais il est déjà très performant avec une seule instance où l'on manipule plusieurs millions de nœuds et de relations. Neo4j reste pour le moment un projet un peu en retrait au sein du mouvement NoSQL mais il y a fort à parier qu'il connaîtra un essor plus important dans le futur si le web sémantique se démocratise car il y est très adapté et pourra y jouer un rôle majeur.

## Quand passer au NoSQL ?

Le tour d'horizon des bases de données NoSQL effectué au cours de cet article a montré une certaine hétérogénéité entre les solutions se réclamant de ce mouvement. Les disparités concernent en premier lieu le mode de représentation des données. En effet, chaque grande famille NoSQL apporte des réponses à des cas d'utilisation spécifiques liés aux données manipulées. La première raison qui doit guider une équipe de développement dans son choix

de passer au NoSQL concerne donc la donnée. Ainsi, si l'application à développer doit gérer un modèle de données hyper hiérarchique, il sera tout indiqué de se tourner vers une base de données graphe telle que Neo4j. Il en va de même pour les autres modèles NoSQL définis dans cet article. L'autre avantage, non négligeable, de ces solutions concerne la souplesse qu'elles amènent dans la gestion du modèle de données puisque celui-ci n'est jamais figé et qu'il peut évoluer constamment sans aucune difficulté. NoSQL nous pousse ainsi à nous intéresser de plus près à la donnée et surtout à la manière dont notre application va l'utiliser.

Une fois ce critère pris en compte, NoSQL devra être considéré pour des applications distribuées devant manipuler de très gros volumes de données tout en ayant à répondre à des problématiques de charge et de disponibilité très élevées. En effet, avec certaines solutions NoSQL on atteindra des niveaux de scalabilité très intéressants à des coûts bien plus bas qu'avec des SGBDR.

Le choix d'une base de données NoSQL, et tout d'abord d'une famille de base NoSQL, devra tenir compte de ces 2 paramètres et en cela, la figure 5 donne un bon aperçu du compromis qu'offrent ces familles par rapport à ces 2 critères. Ce schéma, adapté d'une présentation de la société qui est derrière Neo4j, reflète assez bien la situation.

Malgré l'enthousiasme que peut générer le mouvement NoSQL, il faut également garder à l'esprit le fait que ces technologies sont très jeunes et de fait loin d'être matures comparées à des SGBDR tels que MySQL ou PostgreSQL qui ont fait leurs preuves depuis un grand nombre d'années. En outre, les communautés les entourant sont plus réduites, ce qui obligera les développeurs les utilisant à s'y investir réellement, voire à mettre les mains directement dans le code de ces solutions pour les améliorer et corriger d'éventuels problèmes. Enfin, le manque d'un standard pour l'utilisation de ces solutions pourrait être un frein, même si celui-ci s'explique par la disparité de ces solutions notamment. Il faut toutefois noter que SpringSource est en train de se positionner dans ce créneau avec le projet Spring Data qui propose un framework composé de templates de programmation, très simples pour le moment, pour certaines de ces bases NoSQL.

## Conclusion

NoSQL est un mouvement en pleine expansion qui doit être vu avant tout comme une alternative aux solutions proposées par le monde relationnel et non comme la nouvelle "silver bullet" des bases de données qui va remplacer les SGBDR. Ces alternatives devant être employées dans les cas d'utilisation pour lesquelles elles ont été prévues au risque d'avoir de mauvaises surprises au niveau performances. Prometteuses, les technologies qu'il regroupe bénéficient d'un statut un peu particulier pour des solutions émergentes puisqu'elles sont déjà reconnues pour leurs performances et leur fiabilité du fait qu'elles sont utilisées avec succès. Cependant, les besoins de ces sites sont plus que hors normes comparé à ce que l'on peut connaître au sein des applications d'entreprise. De fait l'adoption en entreprise de ces solutions n'est peut être pas encore pour demain mais il y a fort à parier que ce mouvement va continuer à prendre de l'ampleur dans les années qui viennent.

■ Sylvain Saurel – Ingénieur d'Etudes Java / JEE  
sylvain.saurel@gmail.com

# NoSQL dans un contexte Windows Azure

NoSql fait son chemin dans les projets Web et cloud, alors où en êtes-vous et comment aborder ce modèle sans se faire mal ?

**A**vant de nous lancer dans le vif du sujet voici la typologie qui commence à faire l'unanimité :

## Clé-valeur

Ce modèle est le plus simple à aborder : imaginez que vous travaillez sur un cache réparti, avec les mêmes forces et contraintes qu'un cache (requêtage quasi-inexistant en dehors de la clé, données opaques pour la base et donc non requêtables hors code). Nous évoquerons AppFabric Caching, qui reprend ce modèle.

## Colonne

Au-delà du modèle Clé-Valeur, dans le modèle colonne, la base a un peu plus de connaissance de la structure de chaque enregistrement (chaque « ligne » peut être vue comme une liste de clés et valeurs). Nous verrons plus bas que les tables Azure non relationnelles sont un exemple de ce modèle.

## Document

Souvent présentée comme l'alternative la plus souple aux bases de données relationnelles, c'est aussi celle qui s'en rapproche le plus. On y retrouve souvent un langage de requêtage, des types se rapprochant des types courants, et des index. Les colonnes XML dans Sql Server et Sql Azure donnent une idée des bases orientées document.

## Graphe

Une base de donnée orientée graphe a pour vocation de représenter de la façon la plus intuitive possible les relations qui existent

entre objets, et propose une approche de requêtage centrée sur le parcours de graphe plutôt que d'index et de relations entre entités.

## Application à la montée en charge d'un site Web

Si l'on prend le cas d'une ferme Web classique, la cohérence des données est centralisée dans une base relationnelle [Fig.1].

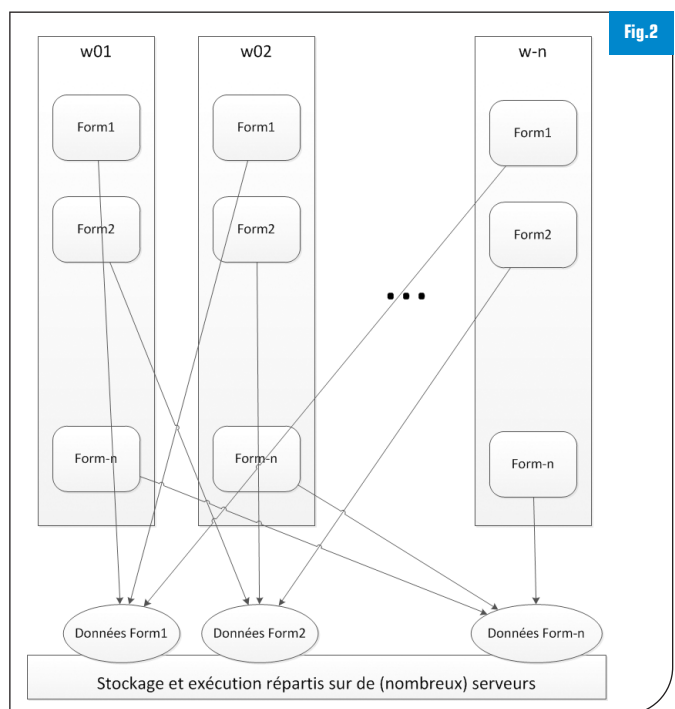
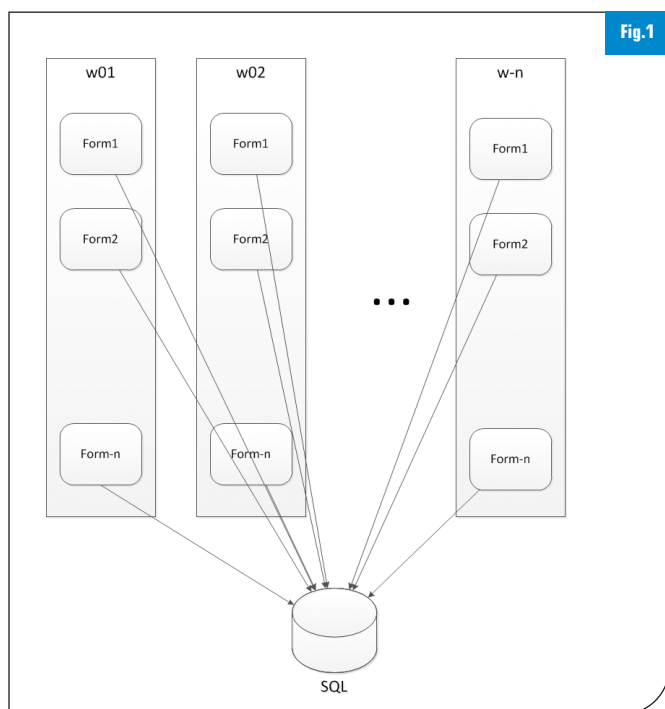
Ce modèle concentre toute la charge de requêtage sur la base, dont la capacité doit croître en fonction du nombre total de serveurs s'y connectant, avec un coût plus souvent exponentiel que linéaire.

Tous les développeurs connaissent la réponse à ce problème : mettre en place un cache entre la base et les serveurs Web. Ce cache a deux qualités :

- Il améliore la localité des requêtes et donc la performance de chaque serveur web
- Il diminue la charge encaissée par la base de données

Ce cache permet également de minimiser l'effort de transformation nécessaire entre le stockage relationnel et l'affichage en stockant les données quasiment comme elles sont affichées, quitte à stocker la même donnée plusieurs fois (le client avec les commandes, mais aussi avec les données détaillées des clients) [Fig.2].

En poussant le raisonnement plus loin, on peut ne plus stocker en base relationnelle tout ou partie des données. En effet, un cache



distribué tel qu'AppFabric Caching permet de stocker la donnée (en mémoire vive) de façon redondante sur plusieurs serveurs. AppFabric Caching, disponible dans Windows Server AppFabric, mais également dans Windows Azure AppFabric (actuellement en préversion CTP), est d'ailleurs conçu pour gérer divers types de données :

- les données de **référence** changent peu (ex : les données de catalogue),
- les données **d'activité** sont en lecture/écriture sans accès concurrents (ex : seule une personne modifie son panier),
- les données de **ressource** sont en lecture/écriture avec accès concurrents (ex : billets de spectacle).

Un cache distribué permet de répondre à un grand nombre d'utilisateurs sur des grandes fermes Web (centaines de frontaux), mais il ne répond pas à la problématique du stockage d'un grand volume de données simplement parce que la RAM est plus chère que le disque. Dans ce cadre, les tables Azure non relationnelles sont une réponse.

### Tables Azure non relationnelles

Il s'agit d'un des services de stockage de la plateforme Windows Azure. La structuration des tables Azure non relationnelles est la suivante [Fig.3].

Chaque compte peut avoir plusieurs tables. Chaque table peut avoir plusieurs entités. Chaque entité a sa clé primaire, son « timestamp » et un ensemble de clés/valeurs. Les entités d'une même table n'ont, par opposition aux bases relationnelles, pas un schéma défini. Les clés et valeurs peuvent varier d'un enregistrement à l'autre (ex : un client a un e-mail, un autre un numéro de téléphone).

La clé primaire composée de la clé de partition et la clé d'enregistrement. C'est l'ensemble des deux qui distingue un enregistrement d'un autre. Par exemple, ci-dessous, le premier et le

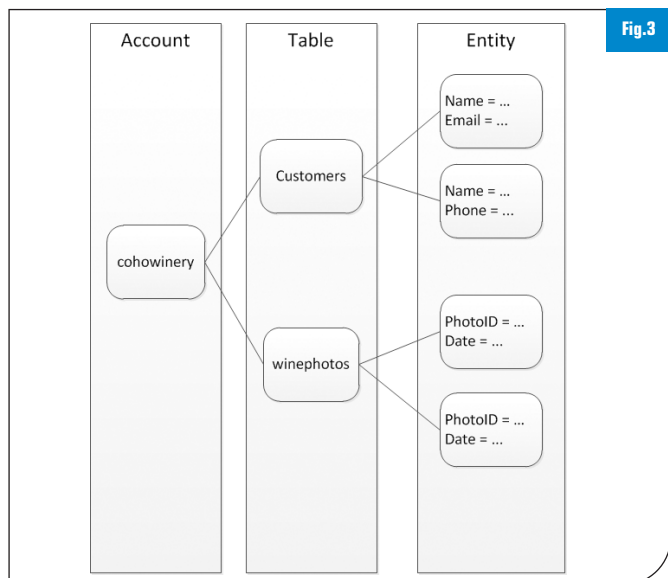


Fig.3

PartitionKey	RowKey	Timestamp	def1	def2	def3	def4	def5
Partition1	rk1	20/10/2010 16:5...	valeur 1	valeur 2			
Partition1	rk2	20/10/2010 16:5...	valeur 3		valeur 4		
Partition2	rk1	20/10/2010 16:5...				valeur X	3.1415926536

Fig.4

troisième enregistrement sont différents bien qu'ils ont le même RowKey [Fig.4].

La clé de partition sert à Windows Azure pour répartir les entrées/sorties (I/O) sur différents serveurs. Si la table est très utilisée, Windows Azure peut décider de faire traiter les I/O de chaque partition par un serveur différent. Si la table est peu chargée, tout peut être traité par le même serveur(1).

Des opérations d'ensemble peuvent être faites sur une même partition ; ce n'est pas le cas sur des partitions différentes. Il est donc important de bien choisir sa clé de partition pour à la fois être capable de répartir les traitements et à la fois avoir un confort de requête correct.

Comme la plupart des services de la plateforme Windows Azure, les tables Windows Azure sont exposées via une API REST. Au-dessus de cette API REST, il existe différentes API de plus haut niveau, comme celle-ci pour Java, pour PHP, et bien sûr pour .NET. On notera que ce dernier SDK permet de requêter les tables Azure noSql via Linq, lui-même syntaxiquement inspiré de SQL !

De façon à aller un peu plus loin, et explorer les possibilités de sélections et d'accès concurrents, on s'appuie sur un exemple de code qui gère de façon simpliste un stock de produits. On crée une table contenant la ligne de produit, l'identifiant du produit, la quantité restante, la date et l'utilisateur qui a généré la dernière mise à jour du stock. Depuis Visual Studio, cette table ressemblera à ce qu'on voit ci-dessous [Fig.5].

PartitionKey	RowKey	Timestamp	AvailableUnits	LastUpdateDate	LastUpdateUser
Aliment	farine001	06/01/2011 15:3...	20	06/01/2011 16:3...	Jean
Aliment	farine002	06/01/2011 15:3...	20	06/01/2011 16:3...	Jean
Aliment	sucre001	06/01/2011 15:3...	11	06/01/2011 16:3...	Jean
Aliment	sucre002	06/01/2011 15:3...	11	06/01/2011 16:3...	Jean
Ustensile	fourchette001	06/01/2011 15:3...	4	06/01/2011 16:3...	Jean
Ustensile	fourchette002	06/01/2011 15:3...	4	06/01/2011 16:3...	Jean

Fig.5

On définit la structure de la donnée dans une classe

```
(...)  
using Microsoft.WindowsAzure.StorageClient;  
using Microsoft.WindowsAzure;  
using System.Data.Services.Common;  
  
namespace AzureTableSample  
{  
    public class InventoryEntry: TableServiceEntity  
    {  
        public InventoryEntry()  
        {  
        }  
  
        internal string ProductLine  
        {  
            get { return PartitionKey; }  
            set { PartitionKey = value; }  
        }  
    }  
}
```

(1) On ne parle ici que des serveurs qui sont en train de traiter les demandes de l'application. Dans tous les cas, chaque enregistrement est écrit sur au moins 3 serveurs différents pour assurer la disponibilité de la donnée.



```

    }

    internal string ProductID
    {
        get { return RowKey; }
        set { RowKey = value; }
    }

    public int AvailableUnits { get; set; }
    public DateTime LastUpdateDate { get; set; }
    public string LastUpdateUser { get; set; }
}

```

Les API .NET prennent en charge la création des tables si nécessaire par l'appel d'une simple méthode :

```

CloudTableClient.CreateTablesFromModel(
    typeof(InventoryDataContext),
    storageAccount.TableEndpoint.AbsoluteUri,
    storageAccount.Credentials);

L'insertion des données s'appuie sur la méthode
public void AddInventoryEntry(InventoryEntry newItem)
{
    this.context.AddObject(InventoryDataContext.InventoryTable
Name, newItem);
    this.context.SaveChanges();
}

```

que l'on appelle comme dans l'exemple suivant :

```

dataSource.AddInventoryEntry(
    new InventoryEntry()
    {
        ProductLine = «Aliment»,
        ProductID = «sucre» + keySuffix,
        LastUpdateDate = DateTime.UtcNow,
        LastUpdateUser = «Jean»,
        AvailableUnits = 10
    });

```

Cela se traduit au niveau HTTP par :

```

POST http://<compte>.table.core.windows.net/InventoryTable HTTP/1.1
(...)

```

De même, on crée d'autres enregistrements, puis on effectue des sélections telles que la récupération de tous les enregistrements, qui correspond dans le code à la requête Linq suivante :

```

var results = from c in context.InventoryTable
select c;

```

La requête Linq est traduite par la requête REST signée :

```

GET http://<compte>.table.core.windows.net/InventoryTable()
qui renvoie

```

```

Aliment-farine001: 20 available, last update: Jean 06/01/2011
15:34:31
Aliment-farine002: 20 available, last update: Jean 06/01/2011
15:34:43
Aliment-farine003: 20 available, last update: Jean 06/01/2011
16:22:35
Aliment-farine004: 20 available, last update: Jean 06/01/2011
16:29:33
Aliment-farine005: 20 available, last update: Jean 06/01/2011
16:34:30
Aliment-sucre001: 11 available, last update: Jean 06/01/2011
15:34:35
Aliment-sucre002: 11 available, last update: Jean 06/01/2011
15:34:46
Aliment-sucre003: 11 available, last update: Jean 06/01/2011
16:27:05
Aliment-sucre004: 11 available, last update: Jean 06/01/2011
16:29:47
Aliment-sucre005: 10 available, last update: Jean 06/01/2011
16:34:30
Ustensile-fourchette001: 4 available, last update: Jean 06/
01/2011
15:34:31
Ustensile-fourchette002: 4 available, last update: Jean 06/
01/2011 15:34:43
Ustensile-fourchette003: 4 available, last update: Jean 06/
01/2011 16:22:35
Ustensile-fourchette004: 4 available, last update: Jean 06/
01/2011 16:29:33
Ustensile-fourchette005: 4 available, last update: Jean 06/
01/2011 16:34:30

```

On peut aussi avoir une requête multicritères telle que :

```

var results = from c in context.InventoryTable
where c.PartitionKey == productLine
&& c.LastUpdateDate >= updatedSince
select c;

```

Qui correspond en REST à :

```

GET http://<compte>.table.core.windows.net/InventoryTable()
?$filter=(PartitionKey%20eq%20'Aliment')%20and%20(LastUpdate
Date%20ge%20datetime'2011-01-06T16%3A33%3A31.2217664Z') HTTP/1.1

```

et renvoie :

```

— toutes les données de type Aliment depuis 06/01/2011 16:
33:31 —
Aliment-farine005: 20 available, last update: Jean 06/01/2011
16:34:30
Aliment-sucre005: 10 available, last update: Jean 06/01/2011
16:34:30

```

**NB :** au niveau du serveur, les colonnes autres que la clé primaire ne sont pas indexées.

On voit donc que les tables Azure non relationnelles supportent divers types de sélections. Elles ont aussi certaines limitations.

Par exemple la requête suivante :

```
var results = from c in context.InventoryTable
               where c.PartitionKey == productLine
                  && c.LastUpdateDate >= updatedSince
                  && c.RowKey.StartsWith(productIDPrefix)
               select c;
est traduite en REST par
GET http://<compte>.table.core.windows.net/InventoryTable()
?$filter=((PartitionKey%20eq%20'Aliment')%20and%20(LastUpdateDate%20ge%20datetime'2011-01-06T16%3A33%3A31.2217664Z'))%20and%20startswith(RowKey,'sucr') HTTP/1.1
mais elle aboutit à une erreur:
<code>NotImplemented</code>
<message xml:lang="en-US">The requested operation is not implemented on the specified resource.
```

En effet, les tables Azure non relationnelles ne comprennent pas un critère de type `StartsWith` (contrairement à une base SQL Azure exposée en OData par exemple). Grâce à la colonne `timestamp`, le moteur peut gérer les accès concurrents, avec une stratégie optimiste. Lorsqu'un client essaie d'effectuer une modification sur un enregistrement en fournissant un timestamp obsolète, le serveur répond ainsi :

```
<code>ConditionNotMet</code>
```

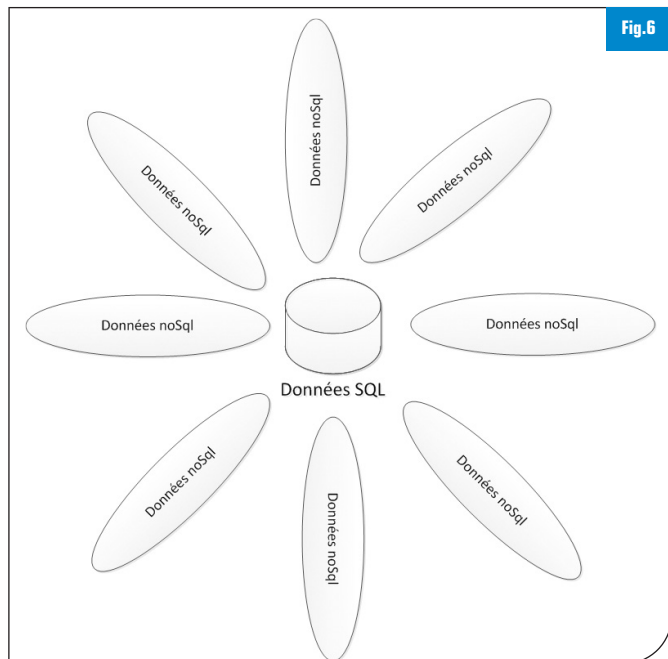


Fig.6

```
<message xml:lang="en-US">The condition specified using HTTP conditional header(s) is not met.
```

On voit donc que les tables Azure non relationnelles permettent

- de gérer les accès concurrents sur une même entité,
- des sélections plus complexes que la simple recherche sur la clé primaire (bien que cette recherche soit plus limitée qu'une requête sur une table unique d'une base de données relationnelle)

En revanche, dès qu'on commence à travailler sur plus d'une table à la fois, les choses tombent vite sous la responsabilité de l'application. Typiquement, on ne peut pas utiliser de clé étrangère. On ne dispose pas non plus de transactions entre plusieurs tables, ni même entre enregistrements ayant des clés de partitions différentes.

Cela peut, dans l'exemple classique des commandes et du stock, se résoudre au niveau applicatif de la façon suivante :

- le module de gestion des commandes ajoute la ligne de commande
- le module de gestion des commandes émet un événement de décrémentation du stock (par exemple dans un flux AtomPub si l'architecture est de type REST)
- le module de gestion des stocks consomme l'événement et décrémente son stock

A aucun moment, on n'a besoin de *commit* à deux phases car on peut retenter chacune des étapes séquentielles ci-dessus sans remettre en cause l'étape précédente.

## Conclusion

Les moteurs noSql tels que les tables Azure non relationnelles ouvrent de nouvelles perspectives, complémentaires des moteurs relationnels, en permettant une montée en charge quasi infinie tant en termes de stockage que de nombre de clients servis, mais cela vient avec de nécessaires changements dans l'architecture de l'application. Ces changements sont trop importants et trop structurants pour être l'unique approche retenue, et l'expérience montre que les qualités apportées par le modèle relationnel classique sont trop importantes pour s'en passer. On panache donc NoSql et Sql en donnant à SQL un rôle concentré sur ses forces. On peut par exemple stocker les données centrales dans SQL Azure, et des données périphériques (qu'on retrouve via une clé commune) dans des bases clé/valeur ou colonne, supprimant ainsi la principale faiblesse de noSql - le requêtage - et obtenir le meilleur des deux mondes [Fig.6].

■ Pierre Couzy & Benjamin Guinebertière

Relations techniques développeurs et architectes,  
Microsoft France.

# Nouveau : économisez jusqu'à 50%

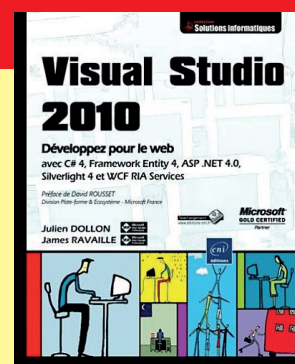
## Abonnement 2 ans au magazine + 1 livre numérique ENI

• **79€ 2 ans au magazine** au lieu de 130,90 (valeur de 22 numéros) Tarif France métropolitaine  
+ un livre d'une valeur de 23,9 € à 31,9 €, soit un total de 154,8 € à 162,8 €

• **89€ 2 ans au magazine + archives sur Internet et PDF + 1 livre numérique ENI**

Livres à Choisir : • Visual Studio 2010 • PHP5.3 • Bing Maps • MySQL 5, Administration et optimisation  
• Java et Spring, Concevoir, construire et développer une application Java/J2EE avec Spring.

Coupon d'abonnement p. 81 et sur [www.programmez.com/abonnement.php](http://www.programmez.com/abonnement.php)



# La mort des bases de données relationnelles et du SQL ?

Et bien devinez quoi, Non ! Nous voilà donc soulagés : pas besoin de toucher à ce sacro-saint SGBDR que l'on aime et que 40 années de bons et loyaux services nous ont permis de maîtriser.

**P**our être franc, NoSQL n'est certainement par le bon acronyme. Alors certes, d'aucuns l'ont fait évoluer en Not Only SQL et c'est bien là le sujet : cet acronyme ne fait que regrouper des solutions applicatives de stockage alternatives à celles usitées traditionnellement (les systèmes relationnels). Ce qui est peut-être nouveau c'est que ces solutions ont été « démocratisées » par les géants du Web que sont Amazon, Google, LinkedIn ou plus récemment Facebook, Twitter et consort. D'abord utilisées en interne pour adresser leurs enjeux, certaines de ces solutions ont été clonées, imitées et offertes à la communauté « en Open Source ».

## Un peu d'histoire...

Fin des années 90, alors que la France croit encore au minitel, la Silicon Valley vit ses heures glorieuses et voit naître – entre autres – deux sites qui deviendront les mastodontes que l'on connaît aujourd'hui : Amazon et Google. Sur la décennie qui suivra, ces sites seront soumis à des courbes d'utilisation peu communes et se devront d'innover techniquement. Google se doit de développer une infrastructure, des modèles de stockage et de programmation lui permettant de « stocker le Web » et de permettre de rechercher de l'information en temps réel. En somme, développer des modèles techniques lui permettant d'analyser des téra et des téra de données. C'est la naissance de **Google Big Table**, **Google File System** et la consécration de l'algorithme **Map/Reduce**. De son côté, Amazon a des enjeux différents : assurer le maximum de disponibilité pour la boutique en ligne mondiale, autrement dit s'assurer la disponibilité du canal de vente... A ce jeu, Amazon.com (via notamment son actuel CTO Dr. Werner Vogels) développe un

système de stockage permettant d'assurer la disponibilité du canal de vente : Dynamo. Reste le seul point commun de ces systèmes : ils sont distribués et souples en termes de structuration du modèle (contrairement à notre modèle relationnel qui lui écrit dans le marbre des tables systèmes les structures des données, leurs relations...).

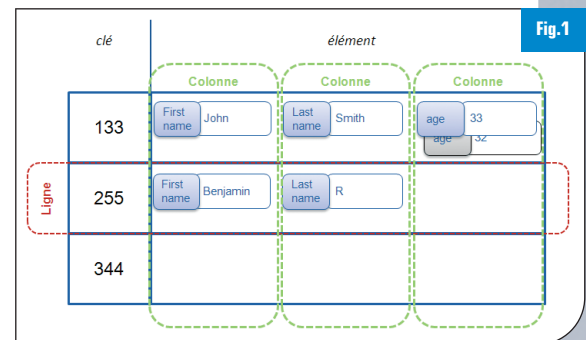
## Le modèle Google : l'« orienté colonne »

Le modèle « orienté colonne » poussé par Google Bigtable touche à la fois à la manière dont est physiquement stockée la donnée sur le disque et également à la modélisation de celle-ci.

Sur le plan de la modélisation, la représentation la plus parlante serait d'imaginer qu'une ligne, un enregistrement, se représente sous la forme d'une table de hachage dont la valeur ou l'élément associé serait lui-même une autre table de hachage. Un enregistrement est donc identifié par une clé (l'équivalent d'une clé primaire) et la valeur est une liste non déterminée de couple clé/valeur [Fig.1].

C'est dans les faits un peu plus vicieux que cela puisque les colonnes sont ordonnancées selon un timestamp (donné au moment de l'insertion). Par exemple, sur le schéma précédent la colonne âge est « multi-valuée » avec deux timestamps qui donnent les âges de 32 et 33 ans.

Les enregistrements sont ensuite regroupés en « familles » (regroupements qui auront des dénominations différentes suivant les solutions techniques) mais qui s'apparentent à nos tables en relationnel. Il est également important de noter qu'il n'y a pas de structure prédéfinie comme nous l'imposent les SGBDR. Les enregistrements au sein d'une même famille peuvent avoir des structures différentes, un nombre



Modèle "orienté colonne"

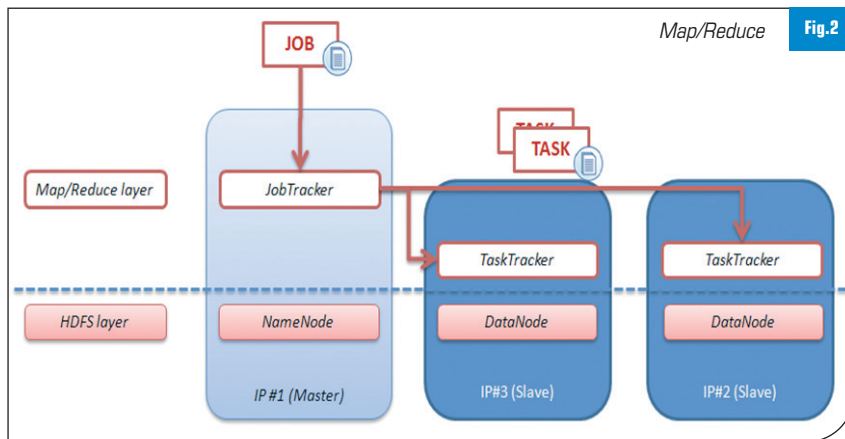
de colonnes variables. En somme, il n'y a pas de schéma prédéfini.

Concernant le stockage physique, le modèle « orienté colonne » - par ailleurs non spécifique à Google Big Table puisqu'on le retrouve dans de nombreux produits commerciaux - sérialise sur le disque les valeurs des enregistrements non pas ligne par ligne mais colonne par colonne. Ce stockage est particulièrement adapté à des utilisations analytiques de la donnée (agrégation...) puisqu'il permet, dans ces cas d'usage, de limiter les accès disque dur (les seeks en fait). N'oubliez pas également qu'il s'agit d'environnements distribués (ie. réparti sur plusieurs machines physiques). Alors pour faciliter le stockage de ces données (ie. la distribution de ces données/fichiers) ainsi que les problématiques de résilience et tolérance à la panne, Google a développé un système de fichiers distribués (ie. capable de fournir une vision unifiée d'un fichier alors que ce dernier est en fait physiquement réparti et répliqué sur plusieurs machines physiques).

## Map/Reduce : le nouveau mode d'interrogation des données...

En termes de requêtage, Map/Reduce devient de facto le nouveau mode d'inter-





rogation des données. Le principal objectif de Map/Reduce est de distribuer le traitement souhaité sur l'ensemble des données (qui rappelons-le, sont stockées de façon distribuée, sur des serveurs physiques différents). En simplifiant, les deux grandes phases d'un algorithme Map/Reduce sont, vous l'aurez deviné :

- La phase « Map ». Le traitement souhaité est distribué sur l'ensemble des machines constituant l'espace de stockage. Chaque machine aura donc à charge d'exécuter le traitement souhaité sur les données localement stockées et de retourner les éléments de réponses correspondants.
- La phase « Reduce ». Le traitement souhaité agrège les différents éléments de réponses issus de la phase précédente pour ne garder, en quelque sorte, que la synthèse des synthèses [Fig.2].

Cet algorithme présente quelques avantages :

- Les traitements ayant lieu pendant la phase « Map » travaillent sur des volumes de données plus faibles. En simplifiant, imaginez que vous travailliez sur 30Go de données réparties sur 10 machines, chaque portion de traitement aura à charge 3Go de données.
- Ces mêmes portions de traitements sont parallélisées, diminuant ainsi le temps de réponse global.
- Traitements et données sont co-localisés. Dans ce modèle, le traitement est envoyé au plus près des données, sur les serveurs physiques.

### Le modèle Amazon : la table de hachage distribuée

Les équipes d'Amazon ont développé, pour leur usage interne, Dynamo ; une table de hachage distribuée.

Dans ce cadre, la modélisation des données est donc encore une fois très simple et peu structurée. Ainsi, tout enregistrement est identifié et identifiable uniquement par sa clé de hachage. Quant aux API d'accès, elles sont simplissimes et ne permettent guère plus qu'ajouter un élément, le requêter via sa clé ou bien le supprimer.

Vous me direz dès lors qu'un tel modèle est stupide, sauf qu'il tire sa puissance de son caractère distribué (donc encore une fois réparti sur plusieurs machines physiques) qui va permettre d'adresser :

- La scalabilité et la tolérance à la panne en écriture en proposant des modèles Master/Master (contrairement aux modèles relationnels qui proposent un modèle Master/Slave)
- L'élasticité de l'infrastructure ou la possibilité de faire varier le dimensionnement de l'infrastructure suivant l'évolution du trafic.

### Partitionnement selon la clé

Partitionner un stock permet de revenir à des volumes de données manipulables mais aussi de distribuer la charge (en espérant une utilisation uniforme de la donnée). Ainsi, le modèle Dynamo propose un partitionnement basé sur la clé (calculé à l'aide de l'algorithme de Consistent Hashing), chaque serveur physique se voit responsable d'un ensemble de clés. Se crée donc une association entre une plage de clés et un serveur. Le principal avantage de l'algorithme de Consistent Hashing réside dans sa stabilité en cas d'ajout ou de suppression de serveurs qui, contrairement aux autres algorithmes de hash utilisant notamment le nombre de serveurs

dans leur calcul, permet de ne recalculer l'association clé/serveurs que sur une faible portion des clés (pour ceux qui aiment les mathématiques, environ  $K/n$  clés ou  $K$  est le nombre de clés et  $n$  le nombre de serveurs). Une stabilité intéressante si l'on souhaite un espace de stockage permettant le redimensionnement de l'infrastructure.

### Privilégier la disponibilité en écriture et être tolérant à la panne

La tolérance à la panne se gère en partie par la réplication des données entre les différents serveurs constituant l'espace de stockage. Ce nombre de réplica  $N$  est configurable en fonction de la donnée.

Mais les solutions telles que Dynamo vont plus loin et veulent privilégier la disponibilité en écriture : ainsi toutes les requêtes d'écriture doivent être acceptées par l'espace de stockage et par n'importe lequel des serveurs le constituant. Dès lors, plusieurs versions du même objet coexistent dans l'espace de stockage. Se pose alors l'enjeu de consistance de la donnée (le « C » de ACID) que ces modèles distribués tendent à délaissier au profit de la disponibilité (noter l'utilisation du verbe délaissier et non pas sacrifier...).

### Une nouvelle gestion de la consistance forte : le « Quorum-Based Protocol »

L'idée derrière la consistance (sous-entendu de la donnée) est que si un client, un programme lit une donnée qu'il vient d'écrire et si bien entendu, il n'y a pas eu de modification entre temps, le client récupère bien la dernière valeur écrite et non pas une valeur obsolète.

Sauf que dans un environnement distribué, le serveur qui servira la requête d'écriture et celui qui servira celle de lecture ne sera très probablement pas le même et de fait la donnée lue ne sera probablement pas la dernière version de cette donnée. En effet, imaginez un espace de stockage composé de plusieurs serveurs. Imaginez maintenant des données « customer » qui seraient stockées sur cet ensemble de serveurs et identifiées par leur numéro de sécurité sociale. Pour être tolérant à la panne, il est décidé que ces données seront répliquées sur un nombre  $N$  de serveurs

(disons  $N = 3$  et nommons les serveurs de A à C).

Si mon client (typiquement une application) modifie un « customer » (cette requête sera effectuée sur un serveur A) puis lit ce même « customer » (cette requête sera effectuée sur un serveur C), il est fort probable que la donnée ne soit pas encore à jour (la réplication n'aura pas eu le temps d'opérer etc.). Imaginons maintenant que lors de l'écriture, cette même donnée « customer » soit répliquée de façon synchrone sur les  $N$  serveurs. Si mon application relit cette donnée, nécessairement et ce, quel que soit le serveur utilisé, mon client récupérera la dernière version de la donnée, garantissant ainsi une donnée consistante.

C'est là le « Quorum Based protocol » : une donnée est consistante si le nombre de serveurs sur lesquels elle est écrite ou lue est supérieur à  $N/2$  ( $N$  étant le nombre de répliques).

C'est également là le génie de ces solutions : il est possible par configuration de faire des « trade-offs » entre la consistance, la performance et la disponibilité, le tout en fonction du type de données que l'on utilise.

### Et NoSQL dans tout cela ?

#### Un écosystème riche, des solutions qui ciblent des enjeux difficilement adressables dans le modèle relationnel

Contrairement à ce que l'on pourrait penser, nous n'avons pas quitté le monde NoSQL puisque à la fois Google Bigtable et Amazon Dynamo sont à la source de la plupart des solutions qui composent ce mouvement.

NoSQL est un écosystème extrêmement riche, varié proposant à la fois des solutions sous la forme de produits (dont vous aurez à charge l'exploitation, le déploiement) mais aussi quelquefois « As A Service ». Alors même si ces solutions partagent des grands principes, « Le diable est dans le détail », et classer ces solutions n'est pas chose simple. Il convient donc d'analyser plus en détail les solutions pressenties et vérifier qu'elles répondent aux enjeux (par exemple à la durabilité qui n'est pas forcément gérée de façon identique suivant les solutions...)

Quoiqu'il en soit, ces solutions permettent d'adresser des enjeux différents :

- *La disponibilité en écriture.* Les

modèles clé/valeur, reprenant une partie des principes d'architectures de Dynamo, comme Voldemort, Riak, Cassandra (que l'on peut classer dans cette catégorie même si sa modélisation est « orientée colonne »), S3 ou SimpleDB en mode « As A Service ». Ces solutions imposent la contrainte d'une faible modélisation et trouveront très certainement une place dans des architectures reprenant les principes d'Event Sourcing (<http://blog.octo.com/en/tag/event-sourcing/>)

- *Analyser des volumes importants de données.* Dans cet univers, le projet Hadoop (clone de Google Bigtable proposant, entre autres, un système de fichiers distribué, un framework Map/Reduce, une base orientée colonne HBase et quelques DSL permettant de s'abstraire de la complexité de Map/Reduce...) est certainement leader. A noter également qu'Amazon offre cette solution technologique « As A Service » via son offre Elastic Map/Reduce. A noter enfin que quelques unes des solutions évoquées précédemment (comme Riak ou Cassandra) offrent la possibilité d'être requêtées via des traitements Map/Reduce.

- *Modélisations complexes ou trop contraignantes dans le monde relationnel.* Cette catégorie contient certes les bases « orientées documents » type MongoDB ou CouchDB qui offrent une certaine richesse de requêtage (colonnes indexables...) sans pour autant imposer la structure normalisée du relationnel. Mais elle regroupe surtout les bases de données graphes – comme Neo4j – qui permettent de modéliser et de faire évoluer des graphes (c'est-à-dire des objets, leurs propriétés et leurs relations) tout en proposant des algorithmes de requête optimisés.

### Et demain ?

Difficile de dire de quoi demain sera fait et dire que NoSQL va supplanter nos SGBDR est de l'ordre du divinatoire et très certainement faux. NoSQL est aujourd'hui un marché jeune voyant émerger de nombreux acteurs qui n'est pas sans rappeler le foisonnement d'innovations d'il y a quelques années autour des serveurs d'applications Java « Open

Source ». Ce qui est plus probable, c'est que NoSQL ne soit que la partie visible d'autres mouvements déjà initiés :

- Les grilles de données massivement utilisées notamment dans les SI bancaires reprennent les mêmes approches et proposent les mêmes « trade-offs »,
- Le monde de la BI traditionnelle s'intéresse de près à Hadoop,
- Les capacités de stockage en RAM de plus en plus importantes poussent à imaginer des stockages mémoire de la donnée (car les temps d'accès sont jusqu'à 100x inférieurs à ceux d'un disque dur) et nécessitent de gérer autrement la durabilité,
- Le « commodity server » a le vent en poupe. Il ne s'agit certes pas de transformer l'ordinateur de votre grand-mère en serveur mais d'imaginer une approche « applicative » du « fail-over ». Autrement dit, privilégier la réplication applicative fournie par ces solutions plutôt que de s'appuyer sur des solutions d'infrastructure type SAN.

Ce qui est certain aussi, c'est qu'il nous faudra apprendre à architecturer nos systèmes en prenant en compte les spécificités de ces solutions (relâchement d'ACID...) et à les exploiter (à moins que le As A Service ne nous simplifie la vie...)

■ Olivier Mallassi

### Références

- [1] <http://nosqlsummer.org/papers> regroupe les principaux papiers fondateurs et dépasse le simple sujet nosql
- [2] Bigtable : A distributed Storage System for Structured Data  
[http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/labs.google.com/fr/papers/bigtable-osdi06.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/labs.google.com/fr/papers/bigtable-osdi06.pdf)
- [3] MapReduce : Simplified Data Processing on Large Clusters  
[http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/labs.google.com/fr/papers/mapreduce-osdi04.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/labs.google.com/fr/papers/mapreduce-osdi04.pdf)
- [4] Dynamo: Amazon's Highly Available Key-value Store  
<http://s3.amazonaws.com/AllThingsDistributed/sosp/amazon-dynamo-sosp2007.pdf>

Il existe de nombreuses autres sources de documentation comme les sites web des solutions qui rentrent plus précisément dans les détails d'implémentation et les subtilités d'utilisation propre à leurs solutions.

# Prévalence : le meilleur de NoSQL

Ou comment une vieille technologie peut se révéler la prochaine grande évolution des systèmes de persistance.

**L**a prévalence est une « vieille technologie » (2001) consistant à réaliser de manière transparente la persistance d'objets Java, .NET, ou de tout autre langage supportant la sérialisation. L'application manipule uniquement les données en mémoire, à l'aide de commandes transactionnelles et elles-mêmes sérialisables. Un système de logs de commande et de snapshots réguliers garantit que même en cas de coupure intertempore, le système pourra être rétabli dans un état consistant. Le fait de router les commandes par un moteur de prévalence réalise ceci de manière transparente, et au final, le développeur a l'impression de ne travailler que sur son modèle .NET ou Java [Fig.1]. Dans la lignée de NoSQL, la prévalence permet donc de manipuler des données en mettant en avant le modèle métier plutôt que le modèle de persistance. Plus de modèle tabulaire, de mapping objet-relationnel, seuls des objets du langage sont manipulés. Par contre, cette technologie va plus loin sur les modèles de données que NoSQL, qui propose typiquement quatre modèles de données, dont le plus utilisé est le dictionnaire de couples clé / valeur. En effet, n'importe quelle structure de classes .NET ou Java peut être manipulée, à la seule condition qu'elle soit sérialisable.

Mais la prévalence hérite aussi de certaines caractéristiques des In Memory DataBases, à savoir que le lieu de résidence traditionnel des données est la RAM, et que la ROM n'est utilisée que comme sécurité pour les accidents majeurs. Une application prévalente est typiquement une application serveur destinée à être accessible en permanence. La présence des données en mémoire offre des performances élevées par rapport à des accès sur un disque dur, y compris les SSD.

## Démonstration

Pour bien comprendre, rien ne vaut du code. Nous allons donc créer un (tout) petit exemple d'application prévalente. Pour cela, commencez par créer avec votre éditeur préféré (personnellement, ce sera Visual Studio) une interface comme ceci : [Fig.2].

Pour fonctionner, un moteur de prévalence a besoin d'une classe de référence sérialisable, et dont il se chargera de garder toujours à disposition du code une instance en mémoire, accessible uniquement par des commandes qu'on lui passera. Cette classe, appelée Referentiel dans notre exemple, contiendra simplement une liste de personnes :

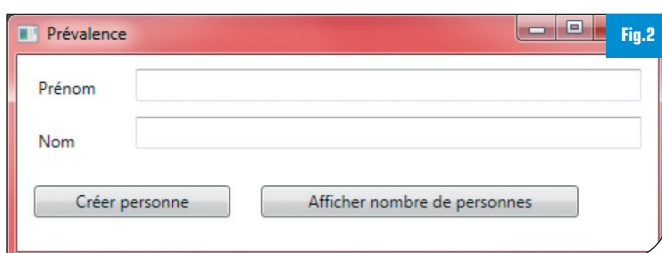
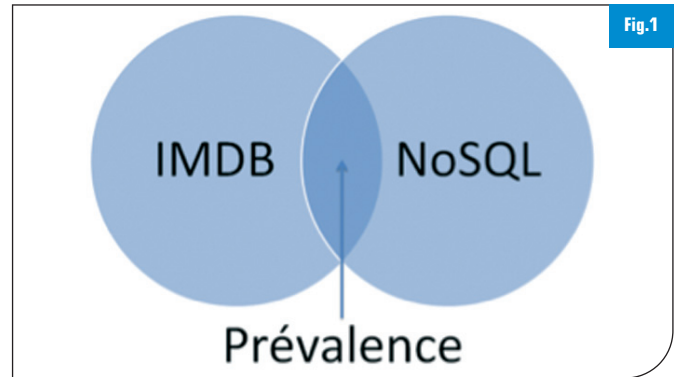


Fig.2



```
using System;
using System.Collections.Generic;

namespace WpfApplication1
{
    [Serializable]
    class Referentiel
    {
        internal List<Personne> ListePersonnes = new List<Personne>();
    }
}
```

La classe Personne est également simplissime :

```
using System;

namespace WpfApplication1
{
    [Serializable]
    class Personne
    {
        public string Prenom { get; set; }
        public string Nom { get; set; }
    }
}
```

Une troisième classe est créée, nommée Moteur, et c'est ici qu'on va commencer à manipuler les concepts de la prévalence. Pour que la solution fonctionne, il faudra référencer la librairie Bamboo.Prevalence.dll (voir lien en référence pour téléchargement).

```
using System;
using Bamboo.Prevalence;

namespace WpfApplication1
{
```



```

class Moteur
{
    private static PrevalenceEngine _Courant = null;
    public static PrevalenceEngine Courant
    {
        get
        {
            if (_Courant == null)
            {
                _Courant = PrevalenceActivator.CreateEngine(typeof(
Referentiel), @"Persistence");
            }
            return _Courant;
        }
    }
}

```

Le moteur de prévalence a été créé en lui passant en premier paramètre le type de la classe dont il doit garantir le stockage asynchrone sur la ROM, et la disponibilité en RAM. Celui-ci étant désormais prêt à être appelé, nous allons créer les commandes de modification ou de lecture des données, en commençant par celle en charge de la création d'une personne dans notre exemple :

```

using System;
using Bamboo.Prevalence;

namespace WpfApplication1
{
    [Serializable]
    class CmdCreation : ICommand
    {
        private string _Prenom;

        private string _Nom;

        public CmdCreation(string Prenom, string Nom)
        {
            _Prenom = Prenom;
            _Nom = Nom;
        }

        public object Execute(object system)
        {
            (system as Referentiel).ListePersonnes.Add(new Personne
()) { Prenom = _Prenom, Nom = _Nom };
            return null;
        }
    }
}

```

Remarquez que la commande est bien une instance, et non une fonction : ses paramètres sont stockés lors du constructeur, et ne seront utilisés que plus tard, lorsque l'exécution sera appelée. C'est ce décalage qui permet au moteur de prévalence de garder la main sur les données, et ainsi garantir qu'il n'y aura jamais de

perte due à une coupure. Les avantages vont d'ailleurs plus loin, car le moteur garantit que la persistance ne sera jamais réalisée tant que la commande n'est pas complètement exécutée. Les commandes étant traitées les unes après les autres, la volatilité de la RAM, souvent un défaut, devient ici la garante de l'aspect transactionnel de l'exécution des commandes.

La commande de lecture du nombre de personnes est plus courte, car elle ne prend pas de paramètres :

```

using System;
using Bamboo.Prevalence;

namespace WpfApplication1
{
    [Serializable]
    class CmdDecompte : ICommand
    {
        public object Execute(object system)
        {
            return (system as Referentiel).ListePersonnes.Count;
        }
    }
}

```

Pour orchestrer tout ceci, il ne nous reste plus qu'à lier le lancement des commandes sur le moteur de sérialisation sur le code lié aux boutons de la GUI :

```

using System;
using System.Windows;

namespace WpfApplication1
{
    public partial class Window1 : Window
    {
        public Window1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, RoutedEventArgs e)
        {
            Moteur.Courant.ExecuteCommand(new CmdCreation(txtPrenom.
Text, txtNom.Text));
        }

        private void button2_Click(object sender, RoutedEventArgs e)
        {
            MessageBox.Show(Moteur.Courant.ExecuteCommand(new Cmd
Decompte()).ToString());
        }
    }
}

```

Vous pouvez désormais lancer l'application, créer quelques personnes, vérifier que le nombre est bien celui attendu. Et surtout, arrêter le processus, le relancer et vérifier que le nombre est le même que lorsque vous êtes sorti du précédent lancement. Où

est physiquement la donnée ? Dans le sous-répertoire « persistance » que nous avons passé en deuxième paramètre, qui contient les fichiers suivants après deux créations de personnes :

Nom	Modifié le	Type	Taille
00000000000000000001.commandlog	22/12/2010 22:58	Fichier COMMAN...	2 Ko
00000000000000000002.commandlog	22/12/2010 22:58	Fichier COMMAN...	1 Ko

Ces fichiers correspondent aux logs des commandes effectuées : lors de l'appel d'une commande, un log sérialisé est stocké avant l'acquiescement de la commande. Ainsi, toute commande acquiescée sera garantie comme exécutée, même s'il y a arrêt du serveur pendant ce temps. De plus, si aucun snapshot n'a été pris, ce sont les logs qui seront rejoués pour ramener l'application dans le même état. On peut le constater en plaçant un point d'arrêt sur la fonction Execute de CmdCreation et en relançant l'application : on verra se ré-exécuter cette fonction autant de fois que de personnes créées. Attention, le chargement du moteur n'aura lieu que lors du premier appel, et il faudra donc par exemple demander le nombre de personnes pour voir ce comportement.

Bien évidemment, ceci poserait un problème pour des temps d'exécution plus élevés que dans le cas de notre exemple. Ou également dans le cas de commande pour lesquels on aurait absolument besoin d'un comportement déterministe. Imaginez le cas d'une commande obtenant par un service web un taux de change : si on la rejoue plus tard, il faudra stocker la date pour être sûr de revenir sur le bon taux. C'est ici qu'interviennent les snapshots : il s'agit d'une photographie de la mémoire à un moment donné, permettant de revenir plus rapidement à cet état lors du prochain lancement, et de garantir la stabilité des résultats des commandes. Pour observer ceci, nous rajoutons le code ci-dessous dans la classe Window1 :

```
protected override void OnClosing(System.ComponentModel.
CancelEventArgs e)
{
    Moteur.Courant.TakeSnapshot();
}
```

Surcharger la méthode OnClosing nous permet de prendre un snapshot mémoire lorsque l'application est fermée. En relançant et en fermant, on constate la différence dans le répertoire de persistance :

Nom	Modifié le	Type	Taille
00000000000000000001.commandlog	22/12/2010 22:58	Fichier COMMAN...	2 Ko
00000000000000000002.commandlog	22/12/2010 22:58	Fichier COMMAN...	1 Ko
00000000000000000003.commandlog	22/12/2010 23:17	Fichier COMMAN...	1 Ko
00000000000000000003.snapshot	22/12/2010 23:22	YourKit Snapshot ...	1 Ko

On peut désormais supprimer les fichiers de logs de commandes, et relancer l'application : elle contiendra toujours les instances de personnes qu'on a créées, et surtout, on ne s'arrêtera plus sur le point d'arrêt fixé précédemment, preuve que les commandes n'ont plus à être rejouées. Il est bien sûr possible de mettre en place des politiques de gestion des snapshots bien plus intelligentes : déclenchées dans le temps, en fonction d'événements, se chargeant de la purge des fichiers de logs, etc. Bamboo est un framework extrêmement complet...

## PLUS LOIN AVEC LA PRÉVALENCE

L'exemple ci-dessus ne rend pas justice à la prévalence : limité par sa taille, il donne une impression très restreinte de la puissance de celle-ci. Pour bien comprendre les apports essentiels de la prévalence, il faut revenir sur quelques points.

### Modèle orienté métier

Tout d'abord, vous pouvez constater que la classe Personne n'a subi aucun impact suite à la mise en place de la prévalence. Nous sommes bien sur des objets POJO, sans aucune adhérence à la technologie de persistance mise en place. On peut d'ailleurs à peine parler de persistance tellement celle-ci est transparente : nous n'avons à aucun moment défini de tables ou de structures quelconques, mais seulement positionné une valeur correspondant à un répertoire.

Au passage, cette simplicité est un atout énorme d'une application prévalente pour les tests d'intégration. Vous avez besoin d'une nouvelle base de données ? C'est aussi simple que de renommer un répertoire. Vous voulez supprimer toutes les commandes lancées sur un scénario de test et revenir à un état initial ? Supprimez les commandes ou recopiez un ancien snapshot. Vous devez déboguer un client sur un enchaînement complexe d'actions ? Demandez lui simplement de vous envoyer le dernier snapshot et les logs suivants, puis progressez en débogage lors de la remontée en mémoire du contexte prévalent.

### Requêtage simplifié

Chaque équipe de développeurs a son « expert SQL »... Pourquoi chaque développeur n'est-il pas en capacité de réaliser des requêtes complexes et performantes ? Parce que SQL est très différent de la POO, principalement. Du coup, imaginez que votre développeur puisse coder ses requêtes en .NET. C'est ce que permet Linq-To-SQL, mais vous devez toujours raisonner en termes de table. Plus besoin de ceci en mode prévalent : vos données sont en mémoire, donc vous pouvez tirer parti de 100% des fonctionnalités de Linq-To-Object.

Un exemple, avec une instruction Linq bouclant sur des réservations, leur appliquant une restriction complexe, puis calculant un barème dont les seuils varient dans le temps, et sommant le tout :

```
decimal SommeMoteur = Reservations
.Where(Res => Resa.UID.Split(new char[] { 'a' }, StringSplitOptions.RemoveEmptyEntries).Count() > 3)
.Select(Res => Resa.Description.Length, Resa => Resa.DateDebut)
.Sum(Valeur => Valeur);
```

Pour fonctionner, il suffit d'une méthode d'extension comme ceci, aisément interprétable par n'importe quel développeur en charge de la maintenance :

```
public static IEnumerable<decimal> Barometre<T>(this IEnumerable<T> liste, Func<T, decimal> BaseCalcul, Func<T, DateTime> DateApplication)
{
    foreach (T Atome in liste)
    {
        int[] Coefficients = new int[] { 0, 4, 9, 12 };
        DateTime DateCoefficients = DateApplication(Atome);
        if (DateCoefficients.DayOfWeek == DayOfWeek.Wednesday) Coefficients = new int[] { 0, 5, 10, 15 };
        else if (DateCoefficients.DayOfWeek == DayOfWeek.Thursday) Coefficients = new int[] { 0, 3, 8, 20 };
        decimal Resultat = 0;
        decimal ValeurReference = BaseCalcul(Atome);
        if (ValeurReference < Coefficients[0]) Resultat = -1;
        else if (ValeurReference < Coefficients[1]) Resultat = 0;
        else if (ValeurReference < Coefficients[2]) Resultat = 1;
        else if (ValeurReference >= Coefficients[3]) Resultat = 2;
        else Resultat = 3;
        yield return Resultat;
    }
}
```

Si vous voulez vraiment vous rendre compte de la puissance expressive de Linq sur un modèle prévalent, essayez juste d'écrire

Supplément  
spécial  
détachable

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

# Tout savoir sur Team Foundation Server 2010



## Cycle de vie des applications

Toutes les nouveautés de TFS  
Créer son premier projet ALM  
Ouverture et interopérabilité vers Java, Eclipse...  
Les développeurs témoignent



# édito

## L'ALM au cœur des projets



La gestion du cycle de vie des applications, ALM(\*) en anglais, constitue un des défis des entreprises, développeurs, et équipes IT. Aujourd'hui, les solutions sont particulièrement matures et ne cessent d'évoluer. Microsoft, avec VS/Team Foundation Server, dispose d'une plate-forme ALM homogène et complète. La version

2010 comble de nombreuses lacunes passées, sur le build, l'ouverture vers le monde non .Net, la collaboration, le reporting.

Avec TFS 2010, les équipes, tous les intervenants à un projet disposent des bonnes données, des bonnes métriques, des bonnes vues pour comprendre, suivre un projet. Le but d'un ALM est aussi bien de concevoir un projet que de communiquer, collaborer, partager. Sans collaboration, le projet n'aura aucune chance de réussir. Sans intégration continue et des builds réguliers, pas de qualité logicielle !

Visual Studio 2010 (TFS étant un des composants de VS 2010) embrasse l'ensemble des fonctions que les équipes, les utilisateurs, les testeurs, les architectes, les développeurs attendent d'un ALM. A la fois agile, par l'intégration de template et orienté modèle, avec pour la première fois le support en standard des diagrammes UML. TFS 2010 apporte un cadre formel de développement, de conception, tout en fournissant tous les outils nécessaires pour définir les process, les itérations, suivre le planning projet, partager les documents, assurer l'intégration continue, les cas de tests.

Comme vous le verrez dans ce dossier spécial TFS 2010 de Programmez !, TFS, de par son architecture modulaire et extensible, s'adapte parfaitement à vos besoins. Et si vous n'êtes pas convaincu concernant les notions de modélisation ou d'industrialisation logicielle, VS 2010 apporte, là aussi, des réponses précises et efficaces au quotidien.

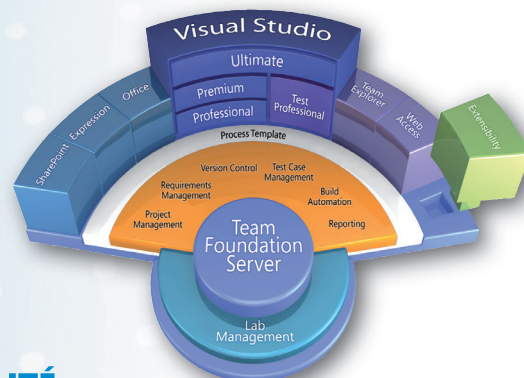
(\*) Application Lifecycle Management

■ François Tonic

# sommaire

## ARCHITECTURE

Les grands principes d'un ALM .....	3
TFS 2010 par le schéma .....	7



## QUALITÉ

Team Foundation Server 2010 : qualité logicielle et agilité .....	8
---	---

## MISE EN OEUVRE

Votre 1er projet ALM avec TFS 2010 .....	12
--	----

## TECHNIQUE

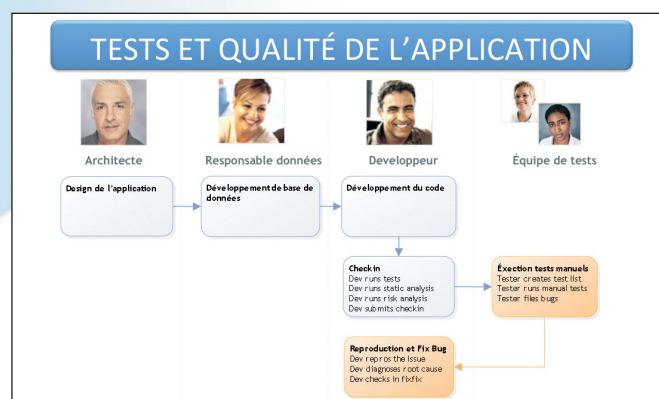
Le besoin d'intégration continue .....	15
Qu'est-ce que Team Explorer Everywhere ? .....	17
Mise en œuvre d'une galerie d'extensions, privée, dans Visual Studio. ....	20

## TEMOIGNAGES

Noham Choulant, Dany Damaj .....	22
Pascal Laforest .....	23

## PANORAMA

A vous de choisir .....	24
-------------------------	----



# Les grands principes d'un ALM

Un ALM n'est pas un simple outil complémentaire. Aujourd'hui, un ALM est à la fois transverse et vertical. Transverse car il doit faire communiquer l'ensemble des départements, des services de l'entreprise (casser l'effet silo). Vertical, car il doit aider les équipes d'un même département, une équipe de développement, à échanger, communiquer entre elles.

Un projet ALM ne s'improvise pas et vous le verrez cette approche bouleverse l'organisation des équipes et de l'entreprise.

Il est tout d'abord important d'auditer et de définir vos besoins, les outils déjà utilisés, le patrimoine applicatif à maintenir (et donc son code) induisant une future migration, définir un organigramme précis des équipes.

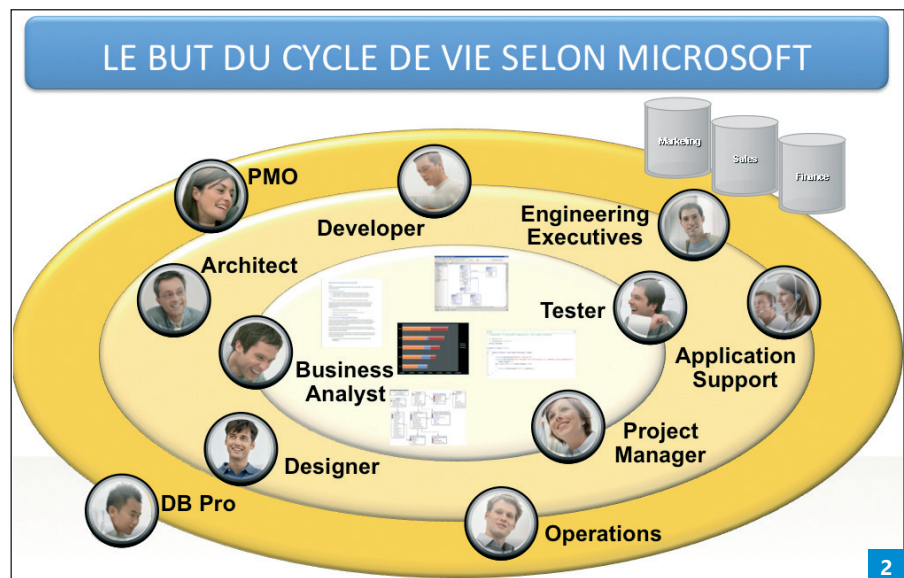
A partir de cet audit, vous pouvez définir les besoins réels et moduler l'ALM en conséquence. Il n'est pas utile de déployer un ALM complet si les besoins se limitent au contrôle de version, à la génération et l'intégration continue.

## Décloisonner l'entreprise, les équipes et introduire la collaboration, le partage

Une entreprise, grande ou petite, possède divers départements (finance, direction, commercial / marketing, informatique...) avec différentes équipes. Bien trop souvent, tout fonctionne en silo avec peu ou pas de communication. L'un des objectifs de l'ALM, qui contribue à la réussite des projets, est de briser cette ignorance et de faire communiquer, collaborer, échanger l'ensemble des équipes, des personnes.

C'est une des clés de la réussite de l'ALM. Pour cela, il faut faire travailler ensemble trois grandes entités : la demande (utilisateur, client, direction, le métier), l'informatique (équipe technique) et la partie déploiement (administrateurs réseaux, responsable projet...).

Complétons ici le point énoncé plus haut : un ALM est à la fois vertical et transverse. Transverse car la gestion du cycle de l'application est une fondation technique apportant les processus, les outils, le reporting, etc. et elle est commune à tout le monde. Vertical car l'ALM s'adresse à chaque personne des équipes avec des outils adaptés à sa fonction, son travail, avec les bonnes informations, la



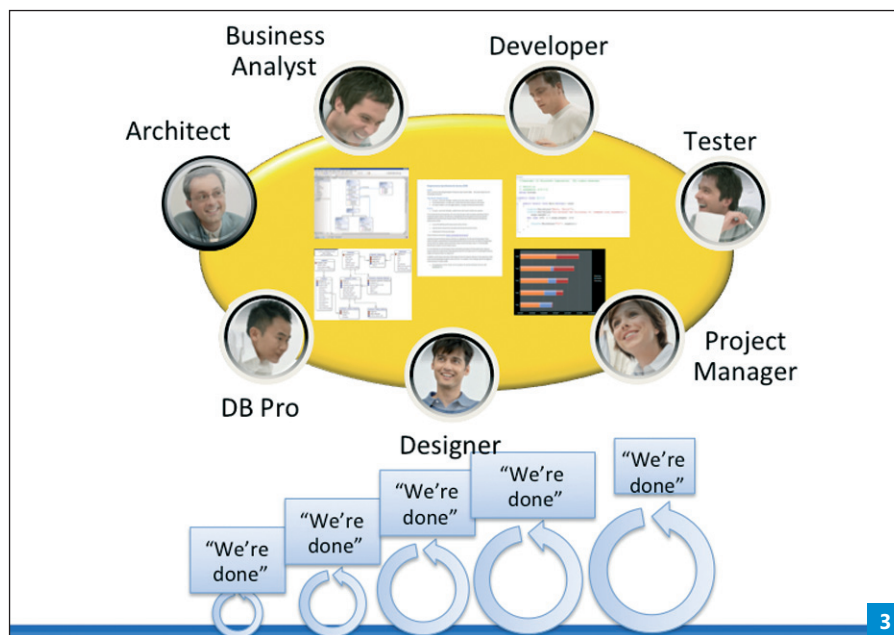
bonne interface. L'ALM est la tour de contrôle qui gère le trafic, les périodes de pointe, les périodes creuses.

## Ne pas tout attendre !

Une solution ALM n'est pas la solution à tous vos problèmes, ni un objectif ultime. Un ALM

ne servira à rien si l'organisation, les méthodes de travail ne changent pas. Et il ne faut pas que votre ALM soit isolé du reste du système d'information. Sa réussite dépend de son intégration et de la bonne volonté de tout le monde !

**Retenez ceci :** un ALM ne s'utilise pas contre



les personnes mais en concertation avec elles. De la gestion et de la conduite du changement (formation, accompagnement des équipes, nouvelles compétences, etc.) dépendront la réussite ou l'échec de la plate-forme de gestion de cycle de vie du logiciel.

## Les grands principes de l'ALM

Une solution ALM est, vous l'aurez compris, à la convergence du gestionnaire de source, de la qualité et des besoins. L'ALM doit être capable de fédérer l'ensemble des outils utilisés dans un projet : contrôle de source, gestion des exigences et des besoins, développement, build, processus et flux projet, gestion de la qualité, intégration continue, déploiement et production, maintenance. Le cycle est sans fin et ne s'arrête donc pas quand le projet est mis en production...

Beaucoup de projets subissent retard et dysfonctionnement suite à une mauvaise gestion des équipes, du développement, à une absence de tests ou tout simplement une mauvaise compréhension des exigences, et des attentes des utilisateurs. Dans le cycle de vie de l'application, il est de plus en plus courant d'appliquer des bonnes pratiques telles que :

- des cycles courts avec des itérations rapprochées (= itérations courtes)
- pilotage du projet par les bonnes pratiques
- focus sur la qualité
- gestion de la maintenance
- collaboration entre les équipes IT et les personnes non IT

Et n'oubliez pas qu'un projet se compose d'un ensemble hétéroclite : utilisateurs, développeurs, chef de projet, architecte, métier, tes-

teur, administrateur (réseau, DBA...). Un ALM mêle aussi bien les aspects humains que les outils **fig. 3 et 4**. Un exemple illustre le défi de la bonne gestion des projets. Comment harmoniser les plannings de chaque intervenant : un développeur fonctionne par jour, semaine et mois (selon les itérations), alors que le métier aura parfois une vision sur 3, 6, 12 mois, etc. Un ALM doit permettre de concilier ces différentes visions. Apporter les bonnes données à chaque intervenant.

## Les piliers métiers et techniques de l'ALM

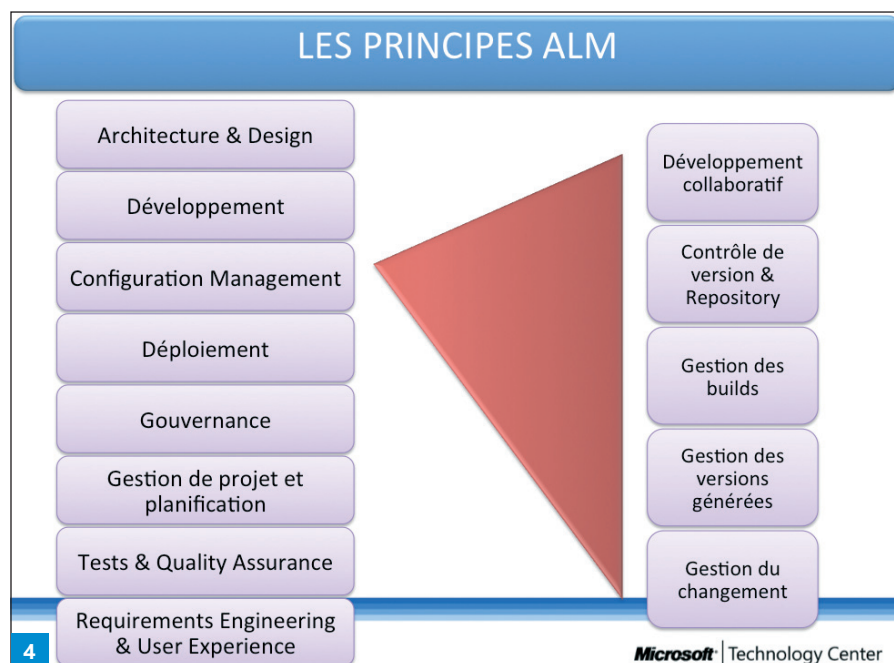
Un outil ALM répond à trois principes : la demande, le développement et le déploie-

ment. A partir de là, on déploie des flux / processus, des outils, des méthodes pour remplir ces trois missions qui recouvrent de nombreuses actions et contraintes. Un ALM fournit un cadre formel de travail et d'organisation.

**1 Planifier et suivre** : récolter les besoins et exigences des utilisateurs. La gestion des exigences constitue un gage de réussite du projet et doit assurer le bon alignement entre les équipes métiers et techniques. A chaque étape du projet (et en particulier lors des itérations), les exigences peuvent changer. L'ALM doit être capable d'intégrer les changements, les injecter dans les processus et gérer les modifications.

**2 Concevoir** : la conception découle de l'expression des exigences. Architecte, chef de projets, utilisateurs et testeurs vont travailler ensemble afin de définir les modèles de données, d'interface, l'architecture du futur logiciel. Cette phase est vitale pour le bon fonctionnement du projet. L'adoption d'une méthode agile, d'un développement par cas de tests et/ou par modèle s'avère souvent utile, voire, indispensable.

**3 Développer** : le développement est l'étape centrale du projet. Il suit les exigences, les modèles définis, les méthodes, les métriques de tests et de qualité. Le projet fonctionnera en itération (cycle de développement) court afin de tester, d'ajuster si besoin, le projet et de corriger les défauts, les bugs au fur et à mesure du développement (bannissez les





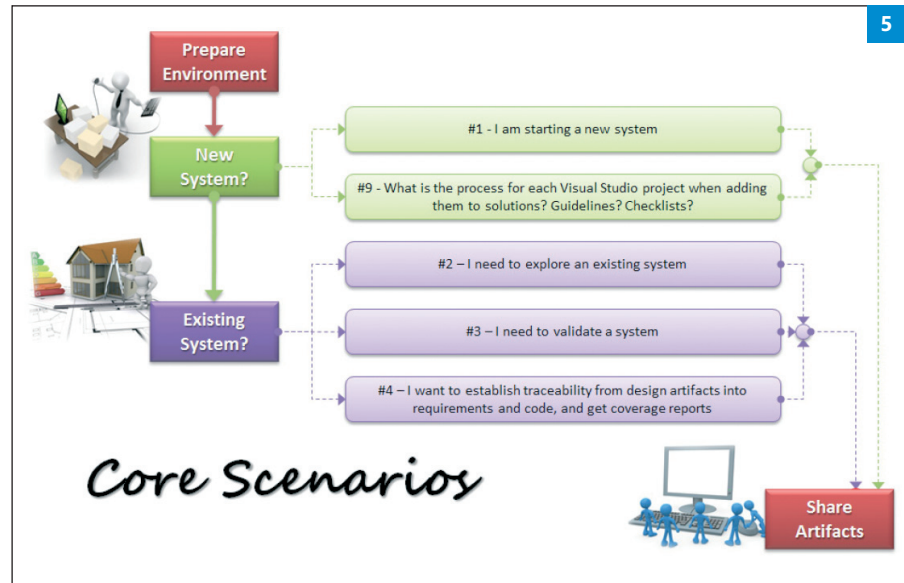
tests et validation des fonctions en fin de développement !). La planification des tâches, les bugs reports, les tableaux de bord permettent un suivi très précis de chaque itération, de chaque membre des équipes. Cela permet de prendre en compte les mesures organisationnelles ou techniques qui s'imposent en cas de dérapage des délais ou dans le développement d'une fonction. La réactivité est primordiale.

**4 Générer :** avec un ALM, la génération des projets se réalise à intervalle régulier que l'on appelle itération. Tous les jours, des scripts de génération et d'intégration continue génèrent le projet en rassemblant l'ensemble des codes. Chaque génération (build) passe par des batteries de tests permettant de vérifier la qualité du code et le respect des métriques et exigences.

**5 Tester :** le test est dans les gènes de tout ALM, de toute gestion de projet. Le test intervient dès le début de la phase de conception. Le test doit intervenir à toutes les étapes du projet. Plus un bug est découvert tard dans le projet, plus il coûte cher à résoudre. La qualité logicielle doit être partout, pratiquée par tous !

**6 Déployer :** l'ultime phase du projet est le déploiement du logiciel. Tout comme le test, le déploiement est testé dans des environnements virtuels reproduisant les plateformes et systèmes cibles sur lesquels fonctionnera l'application. Dès la conception de l'architecture, les systèmes cibles doivent être pris en compte pour ajuster l'architecture et les modèles applicatifs.

Par exemple, le déploiement sur des terminaux mobiles diffère d'un déploiement desktop et le développement nécessitera des ajustements spécifiques. Mais la vie du projet ne s'arrête pas au déploiement. Il faut prendre en compte la maintenance. Le cycle de vie d'une application est un cercle sans fin. Car à chaque demande de maintenance, le cycle se relance.



## Le référentiel

On peut définir trois groupes dans le paysage applicatif : ceux qui demandent (équipe, direction, utilisateurs), ceux qui développent (développeur, chef de projet, testeur, Scrum Master, etc.) et ceux qui déploient (administrateurs, responsable réseau, etc.).

Un ALM doit fédérer l'ensemble des outils, des données spécifiques à chaque personne, à chaque groupe dans un référentiel central. Ce référentiel est le cœur de l'ALM. C'est sur lui que reposent les outils et donc les équipes. La présence du référentiel central est essentielle dans la gestion centralisée des différents assets projets et des codes sources, même si l'infrastructure de son ALM est déployée sur différents serveurs physiques. Ce référentiel s'appuie sur une base de données. Si l'ALM est déployé sur plusieurs sites, il est possible d'avoir un référentiel par site mais toujours doté d'un référentiel central qui s'occupera de l'intégration continue, des builds finaux. Un ALM mal calibré vous coûtera cher pour une efficacité douteuse.

## Ne pas se focaliser sur le coût

Le choix d'un ALM se pose aussi en termes financier. Une solution de gestion de cycle de

vie représente un investissement en licences, en formation, en gestion et accompagnement du changement.

Sans oublier une planification dans le temps du déploiement et de montée en charge de l'ALM dans les équipes.

Malgré tout, il ne faut pas raisonner en retour sur investissement ou coût de possession, c'est le meilleur moyen de rendre inefficace un ALM. Le cycle de vie est une « infrastructure logicielle » de long terme et son efficacité se mesure projet après projet.

Car un ALM réduira les délais de conception des projets, améliorera la qualité, boostera la productivité, offrira une gestion précise des projets et des équipes.

Mais, un temps de rodage et d'ajustement sera nécessaire pour régler les logiciels, les processus et les équipes. Il faudra aussi prévoir le nombre de licences pour les postes utilisateurs, définir les politiques de sécurité et aligner si nécessaire les mécanismes de sécurité à l'ALM.

## Une question de maturité

Vous l'aurez compris, un ALM touche tout le monde dans l'entreprise : le métier, la technique, la direction. Il permet de fédérer autour d'une colonne vertébrale (ALM) la gestion des projets, la collaboration et le partage d'informations. Mais chaque entreprise étant différente, l'ALM doit s'adapter à vos besoins mais vous devez aussi vous adapter. Par exemple, vous n'utiliserez pas forcément tous les modules de TFS. Vous pourrez choisir ceux qui vous sont utiles.

On peut distinguer trois niveaux ALM selon les équipes :

## A RETENIR

Une solution de gestion de cycle de vie de l'application doit :

- fluidifier les processus, les flux de données, d'information entre les équipes et chaque personne
- aligner le métier / la logique business et le technique
- fournir la bonne information à la bonne personne, au bon moment
- établir un référentiel unique et commun à tout le monde
- créer un cercle vertueux dans la vie du projet de l'expression des demandes à la maintenance

- développeur : on parlera d'usine logicielle / industrialisation logicielle
  - équipe : usine logicielle et méthodologie, avec la gestion projet
  - entreprise : gouvernance, gestion de projet
- La maturité de l'entreprise et des équipes demeure essentielle. Elle doit cependant intervenir à plusieurs niveaux techniques et organisationnels : méthodologie (Scrum, CMMi, processus et flux globaux), gouvernance et gestion de projet, industrialisation logicielle. Vous pouvez utiliser une solution ALM sans utiliser de méthodologie agile. Vous avez le libre choix !

Le moindre projet ALM (quel que soit son niveau fonctionnel) doit être réalisé en concertation avec les équipes et l'ensemble des intervenants de l'entreprise, voire des partenaires externes. **Un ALM ne se monte pas contre les équipes.** Cela passera tout d'abord par un audit (précis) des besoins, car chaque intervenant (développeur, chef de projet, DBA, architecte, direction...) a ses propres besoins et demandes. Il faut pouvoir aligner la solution ALM et la calibrer en conséquence.

Chaque étape doit être expliquée, montrée. L'accompagnement au changement est vital et **incontournable**. Si cette phase n'est pas initiée, l'échec est assuré. Pour être efficace, la plateforme ALM doit être utilisée.

Un des aspects importants de l'ALM est la gouvernance. Elle possède un cycle global et de bout en bout du projet. Elle comporte trois grands outils : la gestion des besoins business, la gestion du portefeuille projet et du portefeuille applicatif.

## Répondre aux besoins : vers l'industrialisation logicielle

Un ALM répond à de nombreux objectifs et besoins :

- combiner les bonnes pratiques, des métriques et des méthodologies,
- automatiser les processus
- couvrir l'ensemble du cycle de vie de l'application : le cercle vertueux
- notion de qualité
- disponibilité de tableaux de bord,

L'ALM doit éviter ce que l'on appelle le développement cow boy : sans ordre, sans direction précise. L'industrialisation logicielle vise à formaliser, structurer le développement des applications en fournissant les outils nécessaires au cycle de vie, la gestion projet, le développement, le test, l'intégration continue, la génération et le déploiement.

L'industrialisation est un tout qui permet, à terme, d'être plus efficace, plus productif et de mieux aligner les besoins métiers de l'entreprise (et donc le « time to market ») avec les équipes techniques. Et réduire ainsi le taux d'échec des projets.

Plusieurs raisons expliquent les échecs des projets :

- mauvaise compréhension des attentes utilisateurs
- architecture logicielle mal définie ou ne correspondant pas aux exigences
- mauvaise gestion de la planification et de la répartition des tâches pouvant provoquer des retards sur les dates

- le cycle d'intégration des différents codes, interfaces, modèles de données étant mal maîtrisé, ou totalement absent, induit des problèmes d'intégration que la gestion du cycle de vie aurait détecté précocement
- découverte trop tardive de dysfonctionnement et d'erreurs d'architectures : coût élevé des réalignements et des modifications

## Améliorer la qualité des projets : itération et intégration continue

Une solution ALM répond à deux (autres) impératifs de la gestion de projets : itération régulière et intégration continue. Ces deux notions sont complémentaires.

L'itération dans un projet est un cycle de développement à période plus ou moins courte qui permet de générer à un instant T le projet. Cette génération peut ensuite être envoyée aux utilisateurs pour évaluation des fonctionnalités, interface, etc.

Chaque itération correspond à des tâches de développements précises que les équipes doivent accomplir selon le planning itératif, quelle que soit la méthode de cycle utilisée (cascade, en V ou agile).

Le projet avance itération après itération. Chaque itération apporte des informations importantes aux équipes : bugs, problèmes de spécifications, etc. Ces demandes seront intégrées dans la gestion des demandes par l'architecte et le chef de projet.

L'intégration continue est l'autre pilier de l'ALM. Il implique tout d'abord l'usage d'un outil de gestion de source (aussi appelé gestion de version ou contrôle de version).

Et le code du projet est partagé et versionné (avec historique, gestion des différentes branches). Les développeurs intègrent chaque jour leurs modifications, rajouts de code.

Des tests d'intégration sont pratiqués en intégrant le code nouveau au reste du code.

En cas de problèmes d'intégration, un reporting est réalisé et le développeur doit effectuer les corrections nécessaires.

A partir de l'intégration continue (qui définit un premier niveau de qualité logicielle), il est possible d'ajuster sa stratégie de génération (build) et d'y effectuer les batteries de tests définis.

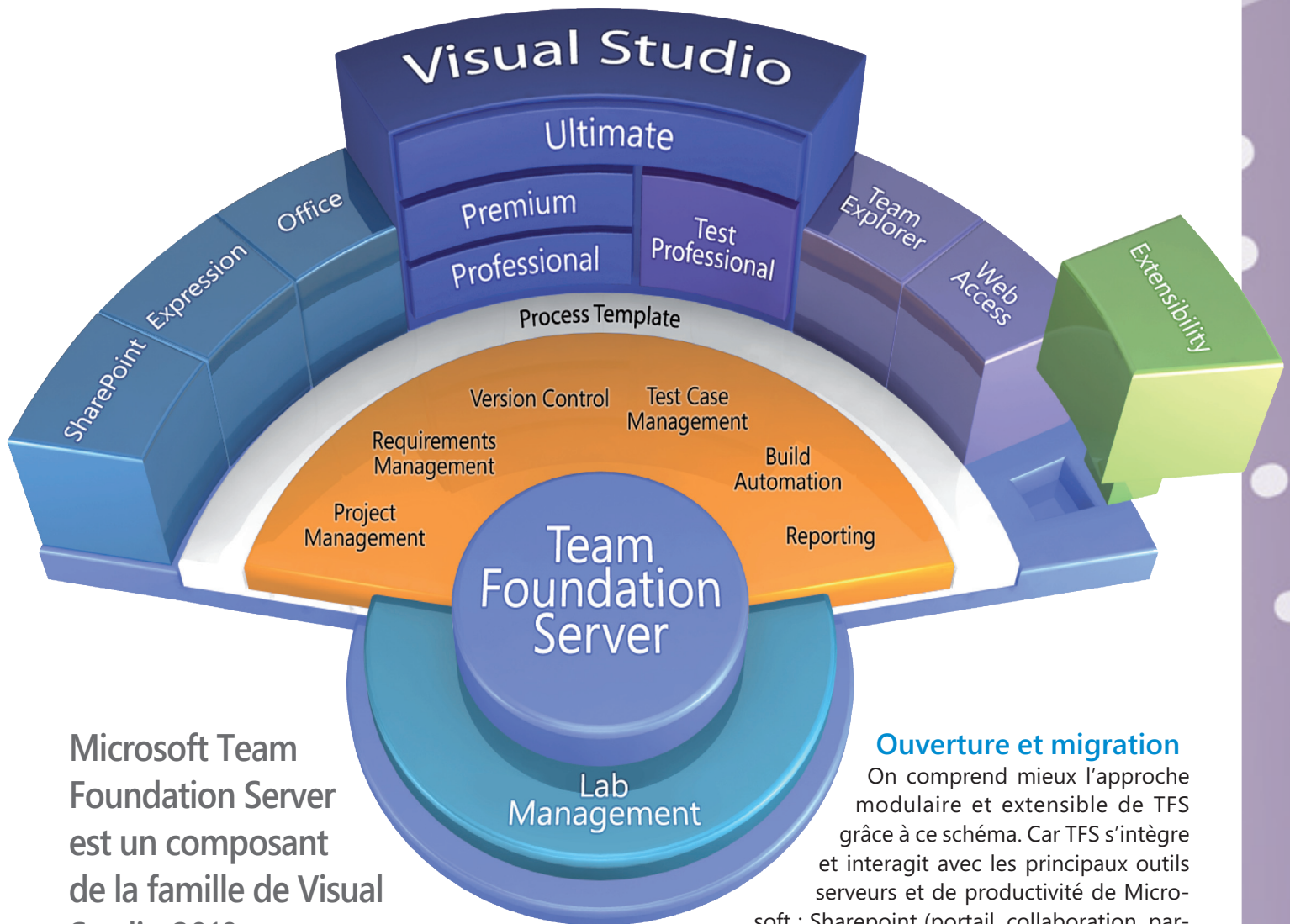
■ François Tonic

## RÉPONDRE AUX CHALLENGES !

Une solution ALM répond à différents scénarios, différents challenges. Retenons-en cinq :

- gestion des projets et des process : avoir des process compréhensibles et plus rapides permet d'être plus productif et aboutir in fine à la réussite des projets.
- Une équipe de développement efficace : le coût annuel de la mauvaise qualité logicielle se chiffre à plusieurs milliards d'euros. Un ALM doit améliorer la qualité du travail des équipes et aboutir à une qualité logicielle satisfaisante
- Assurance qualité et tests : la qualité assurance et les tests garantissent la qualité du projet et améliorent le niveau des logiciels par rapport aux exigences du projet.
- Modernisation, mise à jour et maintenance : ce sont les défis majeurs des logiciels ! Le déploiement et la production d'un projet ne sont que des étapes dans la vie d'une application. La vie du logiciel continue avec les phases de maintenance, de mise à jour et de modernisation (version N + 1).
- Développement hétérogène / architecture hétérogène : aujourd'hui une entreprise n'est plus mono technologie. Les technologies se multiplient et les systèmes se complexifient. Un ALM aide les équipes à intégrer les outils et techniques hétérogènes pour se focaliser sur le projet et sa réussite.

# TFS 2010 par le schéma



Microsoft Team Foundation Server est un composant de la famille de Visual Studio 2010.

Ainsi, les diagrammes UML sont disponibles directement dans l'IDE Visual Studio et non TFS.

## Une approche modulaire

Concernant TFS, la version 2010 fait évoluer l'approche fonctionnelle et l'architecture globale de la plateforme ALM de l'éditeur. Le cœur reste Team Foundation Server qui constitue le moteur de cycle de vie, mais la réelle valeur ajoutée se situe au-dessus, avec la gestion projet et des besoins, le contrôleur de version (au revoir Visual SourceSafe!), les outils de tests, le moteur de build et la partie reporting qui s'appuie sur les derniers outils Excel, Sharepoint.

## Ouverture et migration

On comprend mieux l'approche modulaire et extensible de TFS grâce à ce schéma. Car TFS s'intègre et interagit avec les principaux outils serveurs et de productivité de Microsoft : Sharepoint (portail, collaboration, partage), Expression (design, interface), Office (reporting, planification), Visual Studio (tests et code), Team Explorer et Web Access (portail, accès web, gestionnaire de projet partagé) et enfin l'extensibilité qui se mêle avec l'ouverture. En effet, TFS n'est pas isolé dans le monde Windows / .Net grâce des outils tiers (open source ou commerciaux) ou à TFS Everywhere, TFS travaille avec Maven, Subversion, Mylyn, Ant, Eclipse, PHP, Java... Un bon ALM sait s'intégrer à un univers hétérogène. Cette extensibilité passe aussi par les outils, les assistants de migration de TFS 2008 vers 2010, ou encore en quittant définitivement SourceSafe. TFS sait s'adapter à vos besoins et monter en charge selon vos projets. C'est une des ses forces. Car finalement tout le monde n'utilise pas Sharepoint ou Expression.



# Team Foundation Server 2010 : qualité logicielle et agilité

Aujourd'hui, les tests constituent une étape cruciale de tout projet informatique. Malheureusement, ils sont encore parfois considérés comme quantité négligeable ou variable d'ajustement en cas de retard trop important du projet. Or, le test garantit la qualité du développement et de son bon alignement avec les spécifications dudit projet. Et les tests disponibles sont nombreux : unitaires, manuels, fonctionnels, d'interface, de non régression, couverture de code, développement piloté par les tests, cas de tests...

Il faut distinguer deux types de tests et de testeurs : les généralistes et les spécialisés, comme le montre le schéma suivant : **fig. A**.

Le testeur professionnel a deux objets :

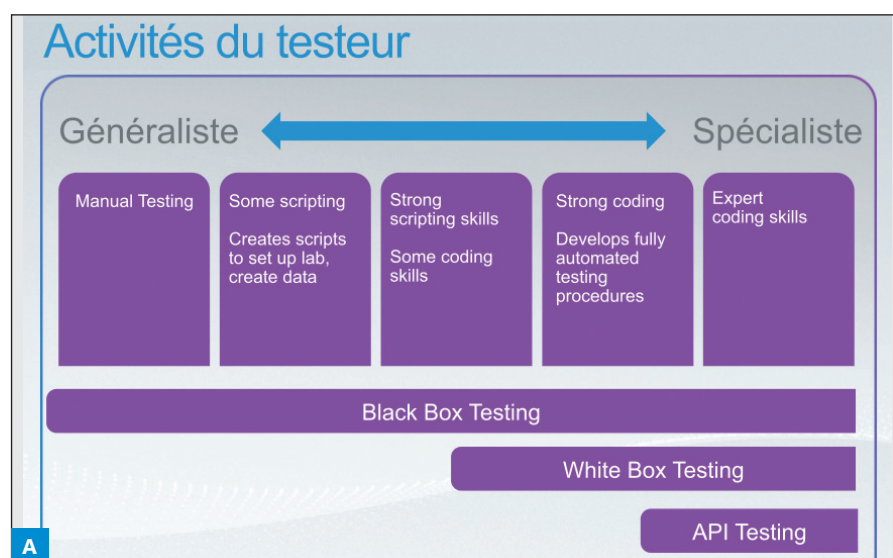
- valider la qualité d'une application
- aider à améliorer la qualité de cette application

Les testeurs interviennent tôt, idéalement, dès la phase de conception, mieux c'est, même si le test n'est pas simple à mettre en œuvre. Car finalement, les efforts de tests peuvent être gaspillés par la répétition de ceux-ci, par les erreurs humaines, le manque de priorité (notamment entre les différents tests). Et les manques d'informations, de contextes de bugs complexes peuvent détériorer les relations entre testeurs et développeurs, mais aussi avec les utilisateurs qui peuvent subir ces bugs à l'usage.

Une des difficultés est de reproduire le bug (= bug reproductible). Ce bug doit être documenté le plus précisément possible, un outil comme IntelliTrace peut aider à suivre la logique du bug. C'est ce que l'on appelle en anglais un « actionable bug ». Test Manager et Test Professional sont là pour vous aider. Ils se chargent des tests manuels, des tests fonctionnels, de l'enregistrement des tests.

## Une gamme 2010 performante : à vous de choisir !

Avec Visual Studio (VS) 2010, Microsoft a rationalisé l'ensemble de la gamme afin de la rendre plus compréhensible. Le tableau suivant présente les tests disponibles dans chaque version de Visual Studio 2010. Pour bénéficier de TFS, vous devez avoir un abonnement MSDN.



Version de Visual Studio	Types de tests
Premium	Test d'impact Profiling de performance Couverture de code Automatisation des tests d'interface Analyse statique du code Métriques de code Tests unitaires de données Tests de génération de données
Test Professional	Idem + Tests manuels Gestion des cas de tests Lab Management
Ultimate	Idem + IntelliTrace Tests de charge Tests web

## Visual Studio Test Professional 2010 : tout pour le test manuel

Cet environnement se dédie aux tests, aux plans de tests, à la collaboration entre développeur et testeur. C'est un outil indispen-

sable pour garantir la qualité. L'outil permet de mieux couvrir le code et l'ensemble du projet tout en améliorant la communication dans les équipes avec la mise en place des procédures et cas de tests.

Il permet aussi de rapidement mettre en place des reportings de tests, avec la gestion des bugs et le taux de réussite des tests. Outil idéal pour réaliser les tests manuels. Surtout, la fonction « enregistrement » permet de détecter les bugs les plus difficiles à traquer (les fameux bugs non reproductibles).

## Visual Studio Lab Management 2010 : le centre nerveux de la qualité logicielle

Dédié aux responsables et équipes de la qualité logicielle, Lab Management est une solution de gestion de « lab virtuel » pour les tests. Il vise à réduire le temps, et donc le coût, de mise en place des scénarii et les batteries de tests. Il s'occupe de créer des environnements virtuels dans lesquels les projets sont buildés

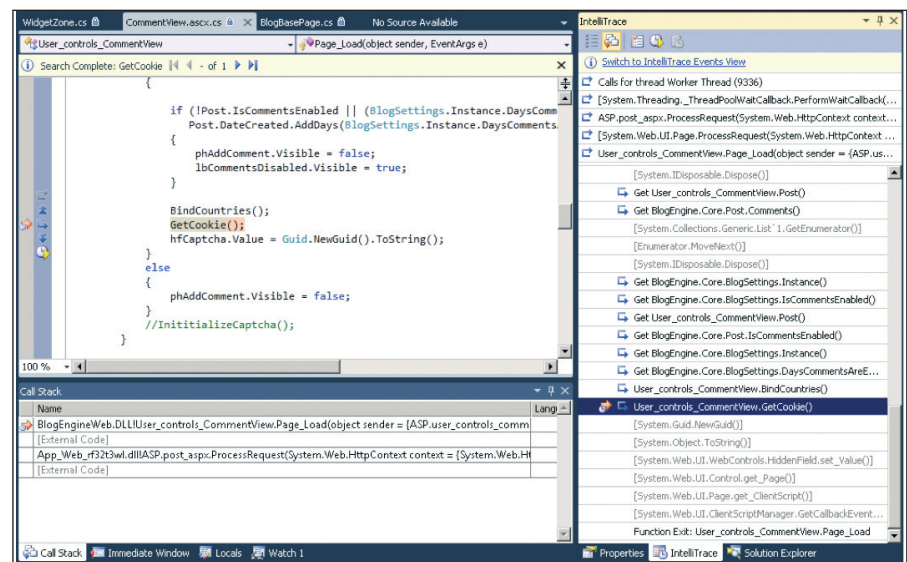
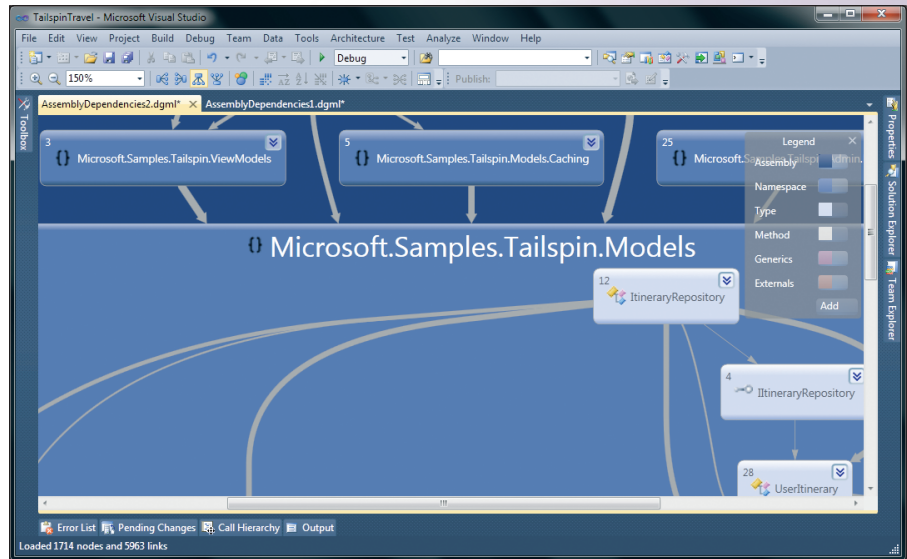
## LES DIFFÉRENTS TYPES DE TESTS DANS VS 2010

- **Tests unitaires** : le plus classique, qui permet de tester du code en .Net, avec un squelette généré.
- **Tests unitaires de base de données** : équivalent du test unitaire .Net mais en T-SQL.
- **Tests ordonnés** : permet d'exécuter un ensemble de tests dans un ordre séquentiel, le test réussit si l'ensemble des sous-tests réussissent.
- **Tests Web** : capture et rejoue un scénario de navigation Web.
- **Tests de charge** : permet de rejouer en boucle un ou plusieurs tests en simulant de la charge pour vérifier les performances de votre application sous différents niveaux de stress.
- **Tests génériques** : test pouvant exécuter n'importe quelle application externe.
- **Tests d'interface utilisateur**.
- **Tests personnalisés** : il est possible de créer vos propres types de tests pour simplifier certaines tâches répétitives (par exemple, la génération d'un proxy lorsque l'on teste des services WCF).

(extrait de « Les Tests logiciels, étude de cas avec Visual Studio 2010 »)

et testés (les images virtuelles fonctionnant sur un serveur Hyper-V). Cet outil vient désormais avec VS 2010 Ultimate et Test Professional. Le processus des tests avec Lab Management peut se résumer ainsi : préparation et création des machines virtuelles du « Test Lab », création des plans de tests. Tout cela se fait dans Test Manager puis tout est stocké dans TFS et au lancement d'un lab de tests, TFS s'occupe de déployer la machine virtuelle, de générer le build, de lancer les plans de tests, de générer les rapports de résultats.

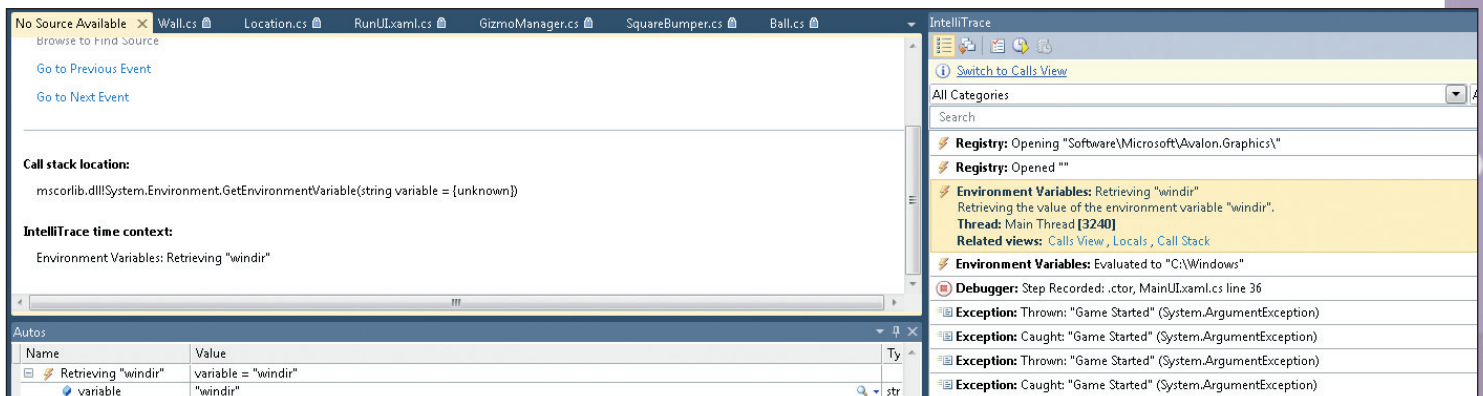
Lab Management est un excellent outil pour les équipes de tests.

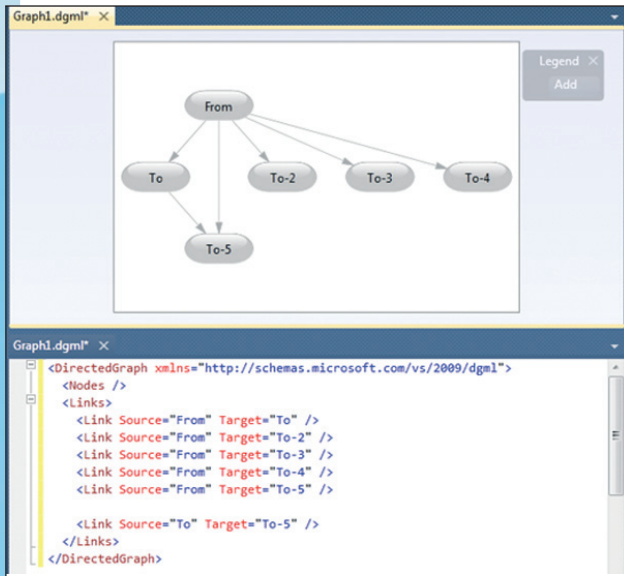


### IntelliTrace : un débogage plus intelligent

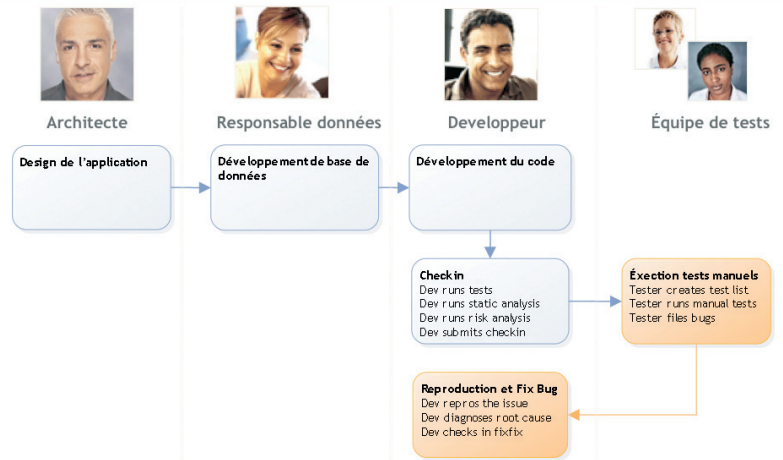
Visual Studio 2010 Ultimate introduit une fonction particulièrement utile pour le développeur et le testeur : IntelliTrace. Jusqu'à présent, le mode debug fournissait un état de votre code à l'instant T. IntelliTrace va au-delà en donnant la possibilité de connaître les événements passés. Très utile pour voir l'impact du code, une régression de code ou détermi-

ner un bug non reproductible. IntelliTrace fonctionne en arrière-plan et enregistre les informations de débogage. Lorsque vous souhaitez consulter un état passé de votre application, vous pouvez passer en mode IntelliTrace. Dans ce mode, vous pouvez accéder à différents instants précis où des événements ont été enregistrés. Vous pouvez également charger et déboguer un fichier journal de débogage de trace doté d'une





## TESTS ET QUALITÉ DE L'APPLICATION



extension .iTrace, créé par IntelliTrace ou Gestionnaire de tests. Les fichiers journaux vous permettent d'exécuter le débogage post-incident avec toute la puissance d'IntelliTrace. Bref, VS 2010 est un centre de qualité logiciel grâce à :

### Test and Lab Management : le test maîtrisé

- 1 • Gestion des cas de test
- 2 • Environnement d'exécution de Tests manuels
- 3 • Provisionnement d'environnements virtuels à la demande

### Débogage, diagnostic et test

- 4 • Test unitaire (code et base de données) et d'interface utilisateur
- 5 • IntelliTrace (« boîte noire »)

- 6 • Analyse, couverture et métriques de code
- 7 • Tests de performance Web et de charge
- 8 • Analyse des tests impactés

### Modélisation des applications

- 9 • Explorateur d'architecture
- 10 • Diagrammes de couche

Pour en savoir plus :

<http://msdn.microsoft.com/fr-fr/library/dd264915.aspx>

## Votre centre de qualité logicielle

La qualité logicielle constitue un point crucial pour l'entreprise et la réussite des projets. Il s'agit de développer, de concevoir un logiciel le plus fiable possible, respectant les demandes et les spécifications. Les équipes de développement respectent des métriques et des règles de programmation, de conception précises et le projet est soumis (très) régulièrement à des batteries de tests et d'intégration continue afin de générer le projet et de détecter les dysfonctionnements le plus tôt possible. La qualité logicielle est possible par l'usage des bons outils.

La maturité des équipes, des entreprises est capitale dans cette notion de qualité, de qualité logicielle. Plus un bug, un problème est détecté tôt dans la conception du projet, moins il coûtera cher (en temps et en argent) à corriger. A contrario, si les tests s'effectuent en fin de cycle, la moindre modification coûtera plus cher car l'impact des changements

sera potentiellement plus important. TFS introduit de nombreux outils et types de tests : manuels, fonctionnels, de charge, stress, impact, cas de test, couverture de test, tests en boîte blanche et noire... Certains sont réalisables par le développeur, d'autres par les testeurs. Le développeur peut les réaliser. Pour des tests automatisés, la création de cas de tests, de projets pilotés par les tests (les tests donnent les résultats finaux que l'application doit fournir et le développeur doit les obtenir), mettre en place des labs virtuels, les testeurs prendront le relais.

Le test d'impact est une aide précieuse pour les équipes. En cas de changement dans les codes, et avant d'appliquer les modifications, vous pouvez observer l'impact qu'elles auront sur le projet, le code. La non régression du code est quelque chose de vital dans le cycle de vie des applications. Trop souvent, des modifications provoquent une régression du code c'est-à-dire que le code perd en qualité et ne répond plus aux spécifications du projet. Une des solutions est d'utiliser des tests d'impacts, en plus de l'intégration continue, des batteries de tests, des builds réguliers.

## LES TESTS IMPACTÉS

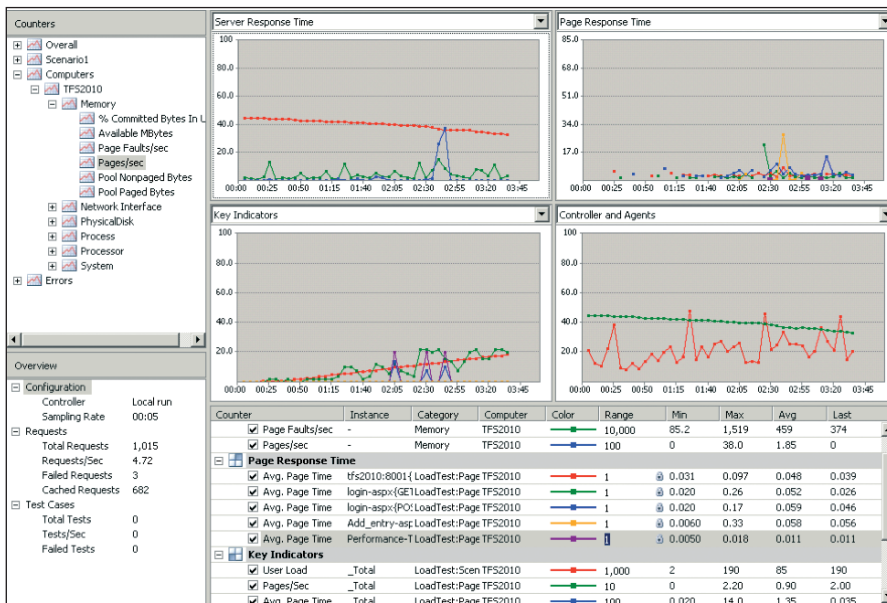
Lors d'une phase de développement, les tests unitaires sont vitaux pour s'assurer que les modifications réalisées sur le code ne donnent pas lieu à un effet de bord sur d'autres éléments de l'application. De ce fait, il est nécessaire d'exécuter les tests le plus souvent possible. Lorsqu'un projet contient un nombre important de tests unitaires, leur exécution peut prendre un certain temps. Pour éviter d'avoir à jouer la totalité de ces tests, Visual Studio 2010 offre une nouvelle fonctionnalité nommée : Analyse de tests impactés. Cette fonctionnalité permet d'enregistrer à chaque test, en exploitant les informations de couverture de code, la liste des blocs de code utilisés lors de l'exécution du test. Grâce à ces informations Visual Studio est capable de trouver la liste des tests impactés par les modifications que l'on vient de réaliser à chaque compilation de l'application. Cette fonctionnalité nommée : « Tests impactés » peut être exploitée localement dans Visual Studio, mais également dans le processus de Build ou encore lors de l'exécution d'un test manuel via l'outil Microsoft Test and Lab Manager.

(extrait de « Les Tests logiciels, étude de cas avec Visual Studio 2010 »)

## Les tests d'interface

VS 2010 introduit une nouvelle forme de test : les tests d'interface utilisateur ou Coded UI Tests. Ces tests sont automatisés et permettent de simuler les actions d'un clavier et d'une souris sur l'interface, à partir de l'enregistrement des actions depuis une exécution manuelle d'un scénario d'utilisation. Pratique et indispensable ! Coded UI Tests s'appuie directement sur le framework UI Automation. Ces tests s'utilisent rapidement via un enregistreur qui capture les actions de l'utilisateur.





## Vous pouvez être agile

Le développement se conjugue avec bonnes pratiques, règles de développement, modélisation et méthodologie et notamment les méthodes agiles (Scrum, eXtreme Programming, Lean...). TFS propose plusieurs méthodologies, agiles ou non, sous forme de templates : MSF for Agile Software Development et MSF for CMMi Process improvement. Agile Software est comme son nom l'indique dédié à une approche agile du projet, du développement. Les grands principes de cette méthode sont : itérations courtes, communication entre les personnes, qualité logicielle, build et déploiement régulier. Cette approche mise sur l'agilité et l'adoption de la gestion du projet et des équipes à des changements rapides et à adopter un développement par cycles courts. On peut le voir comme un mixte des méthodes Scrum et XP. MSF for CMMi, se base sur la méthodologie CMMi qui est une approche très différente car moins souple sur les changements mais qui offre un cadre plus formel et une gestion du projet plus prévisible, tout en ayant comme but ultime : une qualité logicielle optimale. Mais contrairement à une méthode agile de développement, CMMi nécessite une adoption de l'ensemble des spécificités. Avec TFS, votre projet peut utiliser une méthode agile ou non. Vous êtes entièrement libre de votre choix.

## UML et architecture

Le langage de modélisation UML est disponible dans Visual Studio 2010. UML est utilisé par de nombreux outils de modélisation du marché et de nombreux développeurs. Visual Studio 2010 supporte 5 diagrammes UML : diagramme de cas d'utilisation, diagramme d'activité, diagramme de classes, diagramme de composants et diagramme de séquence. Dans Visual Studio, UML permet d'aider au développement logiciel (par la création de modèles), de modéliser les exigences utilisateurs, et enfin de modéliser l'architecture d'un système logiciel. Côté architecture, UML aidera à mieux aligner l'architecture système et applicative aux besoins des utilisateurs. La modélisation permettra de concevoir par les diagrammes une partie de la structure, de

décrire les designs patterns. La modélisation aide aussi à la qualité logicielle en obligeant à une meilleure formalisation des architectures et des exigences, et peut être utilisée dans les cas d'usage.

Les fonctions d'architecture ont beaucoup évolué avec Visual Studio 2010. Ainsi le diagramme de couches facilite la validation d'une architecture (= validation du code par sa conformité au design attendu) et il est possible de lier les classes aux couches définies dans le diagramme de couches.

Autre élément incontournable pour l'architecture, l'Architecture Explorer. Il permet d'appréhender dans sa globalité un système existant, de voir les composants, les modèles utilisés, les connexions entre les composants...

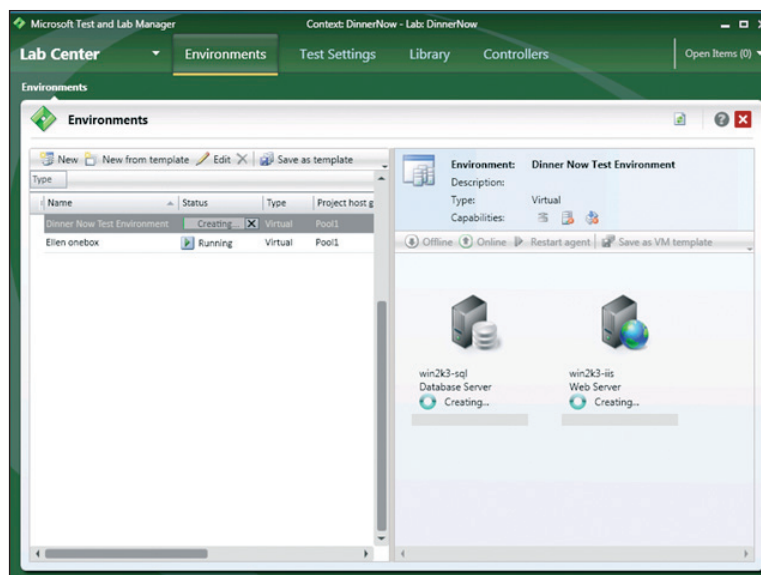
Dans Visual Studio 2010, l'architecture s'articule autour de quatre éléments : DGML, Architecture Explorer, diagrammes UML et Dependency graphes. Dans TFS, tous les diagrammes sont DGML. DGML signifie Direct Graph Markup Language.

Il s'agit d'un langage XML. DGML permet de voir, de comprendre, de visualiser les relations du code projet. Les modèles générés par DGML s'appuient sur le code et le représentent. Ainsi, vous pouvez comprendre l'impact d'une modification sur le reste du code. Le DGML intervient aussi dans l'architecture, en particulier pour créer les diagrammes de couches.

Ainsi vous êtes au plus près du code. Le Dependency graphs est particulièrement utile pour visualiser son code, son organisation, les relations et dépendances. Mais vous pouvez aussi explorer le code via des diagrammes de séquences. C'est là que UML et les modelleurs jouent un rôle important dans votre compréhension du code, et surtout, vos modèles suivent le code au plus près, ce qui est un gage de qualité.

Architecture Explorer est votre tour de contrôle de l'architecture : exploration de l'architecture, représentation du code, créations rapides de graphs depuis une sélection, visualisation des assemblées, types, membres... Pour les architectures complexes, rien ne remplace une vue par un Dependency Graph !

■ François Tonic



# Votre 1<sup>er</sup> projet ALM avec TFS 2010

L'ALM, Application LifeCycle Management, est aujourd'hui une discipline clairement reconnue dans le monde du développement logiciel, mettant en œuvre un ensemble de méthodes et bonnes pratiques clairement identifiées et bien documentées.

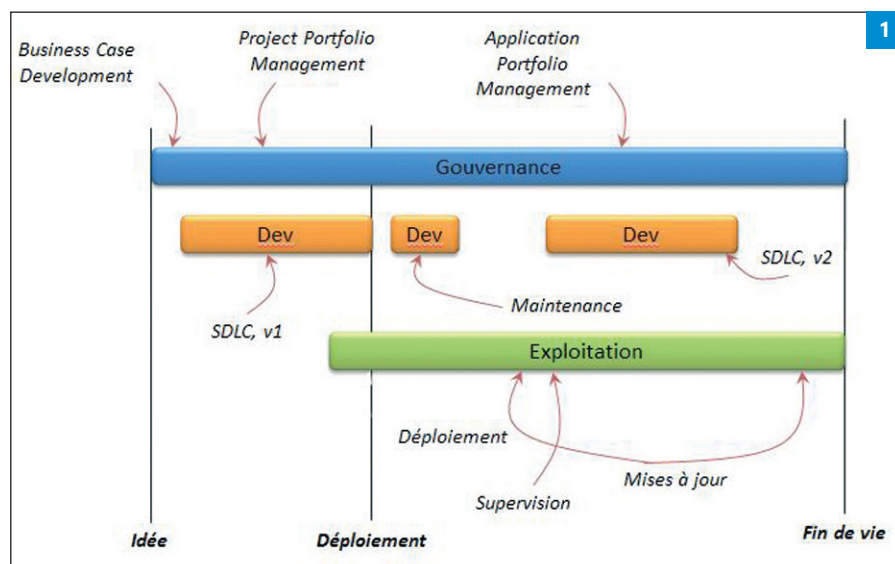
De manière consensuelle, l'ensemble des acteurs du développement logiciel identifie trois grandes phases caractérisant un projet ALM : la gouvernance de projets, la partie développement encore appelée SDLC (Software Development LifeCycle) et la partie mise en production : **fig : 1**. On parle de cycle de vie ou de cycle de développement (SDLC) pour désigner un ensemble d'étapes qui permettent d'aboutir à une application et sa maintenance :

- Définition des besoins ou des exigences
- Conception d'une application
- Développement et production du code
- Mise en œuvre des tests
- Déploiement de la solution

Cette représentation ne doit pas être vue de façon figée, car la mise en œuvre d'une démarche ALM est un processus complexe et qui évolue dans le temps. Le travail d'architecture est-il (doit-il ?) être relié aux tâches de développement ? Les exigences peuvent être techniques ou fonctionnelles ; Dans ce dernier cas on aimerait pouvoir suivre ces exigences tout au long du cycle, jusqu'à l'étape finale où l'on déploie une exigence. Comment relier la phase d'exploitation d'une application à la phase de développement, avec pour objectif, par exemple, de pouvoir identifier précisément d'où provient une anomalie en exécution, aussi bien au sein des exigences que des modules de code associés ?

Enfin, même si cette représentation en cycle semble montrer que les phases s'enchaînent séquentiellement, rien n'impose aujourd'hui une telle démarche : on n'est pas obligé d'attendre la fin de la phase spécifications pour commencer les développements. La mouvance agile et ses méthodes ont planté depuis longtemps un décor plus souple et moins linéaire, facilitant la mise en place de l'ALM dans les équipes.

Les fondamentaux de l'ALM étant maintenant identifiés, nous allons nous intéresser de façon concrète et pragmatique à leur mise en œuvre sur les plateformes de développement modernes :



## «ALM : Keep it simple !».

Ce vieil adage des bons développeurs s'applique aussi aux outils et plateformes de développement qui se doivent d'être le plus pragmatique possible en réponse à la complexité croissante des projets informatiques. La réussite d'un projet passe par un subtil équilibre entre les outils, les méthodes et l'humain (les équipes) **fig : 2**.

Subtil, car il n'existe pas de recette toute faite, immédiatement applicable. Et pourtant, piloter de façon efficace et prévisible un projet de développement ne relève pas de l'utopie : il existe une démarche adaptée et adaptable à votre environnement, tenant compte de vos capacités, des bonnes pratiques mais aussi des « idées » qui ne marchent pas. Et cette bonne démarche, c'est la vôtre, pas celle décrite au fil d'un volumineux livre de méthode : c'est à vous de la trouver, de l'appliquer de façon pragmatique avec un mode opératoire progressif.

C'est là que les outils et solutions Microsoft Visual Studio 2010 et Team Foundation Server 2010 entrent en jeu. Ils couvrent l'ensemble des besoins liés à une démarche ALM, avec un focus particulier sur le socle SDLC (Software Development LifeCycle), vous permettant de

mettre en place votre vision de l'ALM grâce à un paramétrage souple, tout en vous garantissant le respect des grandes lignes générales et des principes fondamentaux de ce type de démarche.


Sans entrer dans trop de détails techniques, nous allons vous donner dans les paragraphes qui suivent les étapes opérationnelles simples pour mettre en place votre 1<sup>er</sup> projet ALM à l'aide de Visual Studio 2010 et TFS 2010. Vous retrouverez dans les lignes suivantes les trois facteurs identifiés ci-dessus, à savoir les humains (tous les acteurs intervenant sur votre projet, les outils et votre méthode).

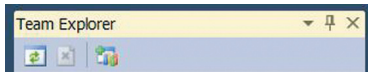
## C'est parti pour un tour complet avec VS 2010 et TFS 2010 :

Nous n'aborderons pas en détail ici la configuration du serveur TFS en lui-même, considérant que votre administrateur vous a donné les informations utiles, à savoir l'URL de connexion, votre login et votre mot de passe. Pour information, le panneau de configuration TFS ressemble à l'écran **fig : 3**.

### Revenons dans Visual Studio 2010.

La 1<sup>ère</sup> étape consiste à connecter votre VS 2010 à votre serveur de développement TFS

2010. Pour cela rien de plus simple : Lancez VS 2010, dans le menu **View**, sélectionnez **Team Explorer**. Ce panneau qui s'ouvre sera ensuite votre point d'entrée connectant votre VS 2010 au serveur TFS. Vous y retrouverez par la suite tout ce qui régit votre démarche ALM, de la documentation au paramétrage en passant par les outils collaboratifs. Effectuons cette connexion au serveur de développement : dans ce panneau **Team Explorer** que vous venez d'ouvrir, cliquez sur l'icône  afin d'ajouter et paramétrer une connexion vers votre serveur TFS :



Dans la suite d'écrans qui s'affiche, cliquez sur le bouton « **Servers** » puis « **Add...** », afin de compléter les différents formulaires proposés : Vous devrez indiquer votre adresse IP ou l'URL de votre TFS (vous pouvez laisser les autres paramètres par défaut), un login et mot de passe (à demander à votre administrateur TFS). Sélectionnez ensuite la collection par défaut **DefaultCollection**, et cliquez sur **Connect**.

Quelques instants de patience... Félicitations : vous êtes désormais connecté à votre serveur TFS ! Ce panneau Team Explorer sera désormais votre point d'entrée pour tout ce qui concerne votre serveur de développement TFS. Vous pourrez toujours le récupérer depuis le menu **View** en sélectionnant le choix **Team Explorer** **fig : 4**.

A ce niveau, vous avez maintenant accès aux différents menus de paramétrage de votre serveur TFS. En cliquant sur le bouton droit, vous accédez au menu contextuel de configuration de votre serveur. Vous avez les possibilités suivantes depuis le menu « Team Project Collection Settings » :

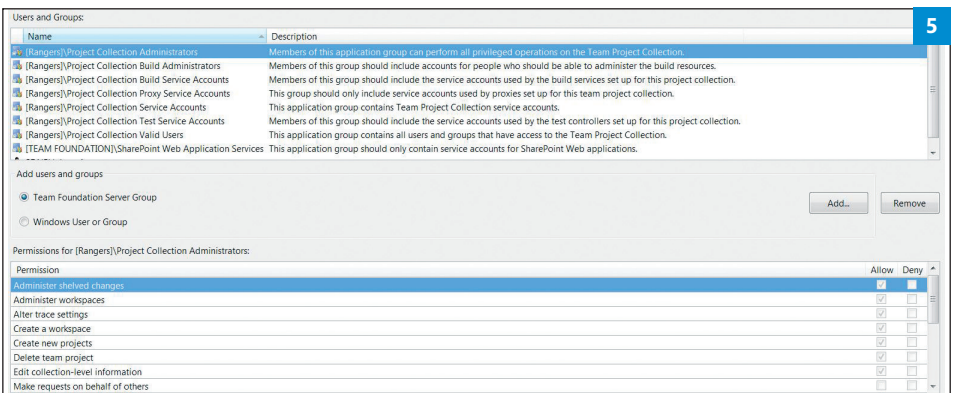
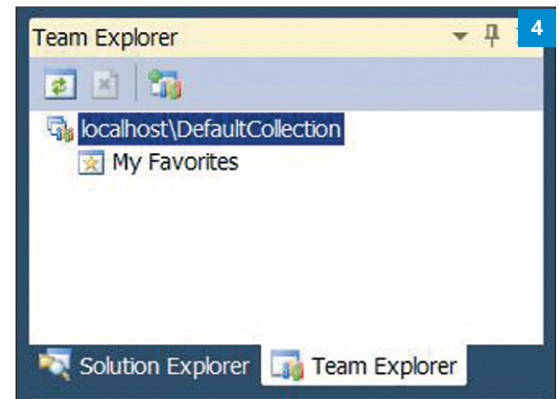
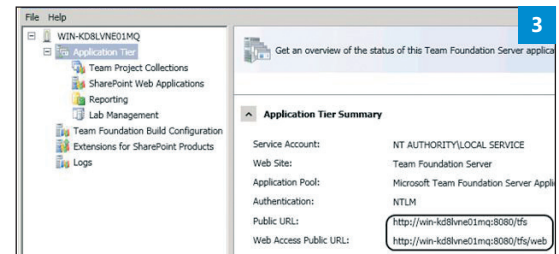
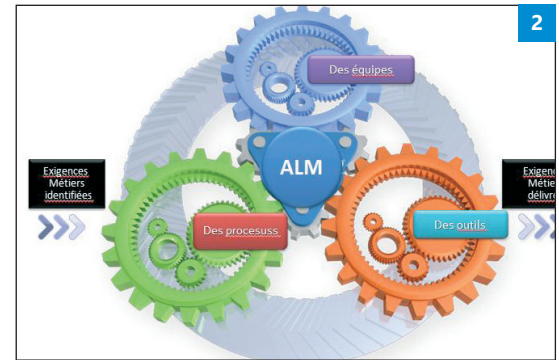
- Configurer les permissions et groupes des personnes autorisées à se connecter à votre serveur. Vous jugerez certainement opportun de bien configurer ces droits d'accès en fonction des profils des personnes utilisant votre serveur TFS (architectes, développeurs, gestionnaires de build, etc.). Ces profils ou groupes d'utilisateurs peuvent être stockés au choix dans TFS ou provenir de vos comptes et groupes Windows.

La granularité des autorisations est assez fine, puisque vous avez la possibilité de donner accès ou non à un grand nombre de fonctionnalités, comme le montre l'écran ci-dessous : **fig : 5**.

- Configurer l'utilisation du gestionnaire de contrôle de sources en déclarant les types de fichiers et leurs extensions.
- Gérer les différents « Process Template » présents sur votre serveur. C'est là un des dispositifs clés de votre démarche ALM, car un process template peut être vu comme l'implémentation de votre méthodologie dans le couple VS/TFS ! Une fois extrait de TFS, un Process Template contient dans un formalisme XML un grand nombre d'informations avec, entre autres, les work items de base, ceux que vous souhaitez voir dans tous vos projets, la gestion des documents et du portail documentaire de vos projets, les aspects reporting (pilotage par tableaux de bord), la configuration du moteur de build, etc.

Voici le contenu « à plat » d'un process template, une fois extrait de votre TFS ; c'est un ensemble de dossiers et de fichier XML que vous pouvez éditer et modifier : **fig : 6**.

Pour vous permettre de travailler plus facilement et efficacement, vous téléchargerez sur le site MSDN le « Process Template Editor » contenu dans les



Nom	Modifié le	Type
Build	19/10/2010 17:27	Dossier de fichiers
Classification	19/10/2010 17:27	Dossier de fichiers
Groups and Permissions	19/10/2010 17:27	Dossier de fichiers
Reports	19/10/2010 17:27	Dossier de fichiers
Test Management	19/10/2010 17:27	Dossier de fichiers
Version Control	19/10/2010 17:27	Dossier de fichiers
Windows SharePoint Services	19/10/2010 17:27	Dossier de fichiers
WorkItem Tracking	19/10/2010 17:27	Dossier de fichiers
ProcessTemplate.xml	21/01/2010 17:45	Document XML



Power Tools. Point important, nous parlons bien ici de VOTRE méthodologie : vous pouvez utiliser directement un des process template proposés en standard (CMMI ou MSF Agile), mais il y a de fortes chances que le bon process template soit une déclinaison de l'un de ces modèles, adaptés à votre besoin spécifique. C'est la grande force de TFS 2010 : vous pouvez éditer et adapter votre propre process template à vos besoins et faire évoluer ce dernier en fonction de votre maturité : c'est l'ALM à la carte ! Il en existe de nombreux sur le marché, certains orientés spécifiquement vers les méthodes agiles comme SCRUM, d'autre dédiés aux tests logiciels par exemple. Ne négligez pas cette étape de configuration : elle est aussi importante pour le bon déroulement de vos futurs projets que la bonne installation et configuration de vos serveurs TFS. Votre VS est donc maintenant connecté à votre serveur TFS, ce dernier étant configuré de façon à suivre votre démarche ALM.

Il vous reste maintenant à créer votre 1er projet d'équipe sur le serveur. Attention, on parle ici de projet d'équipe, pas de projet de développement contenant vos codes sources. Pour cela, toujours dans ce même panneau Team Explorer, cliquez sur le bouton droit sur la collection par défaut, puis **New team Project**. Laissez-vous guider ensuite en sélectionnant les valeurs par défaut aux différentes étapes. Donnez un nom à votre projet d'équipe, sélectionnez par exemple la méthode MSF Agile (étape appelée Process Template) ou sélectionnez à ce niveau votre propre Process Template, celui que vous avez-vous-même inséré dans votre serveur TFS (cf. ci-dessus) et qui correspond à votre démarche ALM précise. Au final vous devriez après quelques instants vous trouver en face du contenu suivant dans **Team explorer** : **fig : 7**.

Ce projet d'équipe contiendra tous les éléments utiles à la gestion du projet, ses utilisateurs, ses règles de sécurité et autorisations, les documents du projet, la définition des tâches, le suivi par tableaux de bord, etc. Ces

éléments proviennent de la configuration de base enregistrée par vos soins sur le serveur (étape précédente) mais peuvent aussi être configurés en cliquant sur le bouton droit de votre Team Project qui vous ouvre un nouveau menu contextuel, vous permettant d'affiner pour ce projet les groupes et permissions, mais aussi de personnaliser vos politiques de check-in et check-out avec vos règles personnalisées.

A cette étape, il est important de bien vérifier dans vos « Team Project Settings », la définition de vos « Area » et « Iterations » : **fig : 8**.

De façon usuelle, « Area » qualifie les différents modules de votre solution (DAL, UI, BL par exemple) et les « Iteration » les différentes versions.

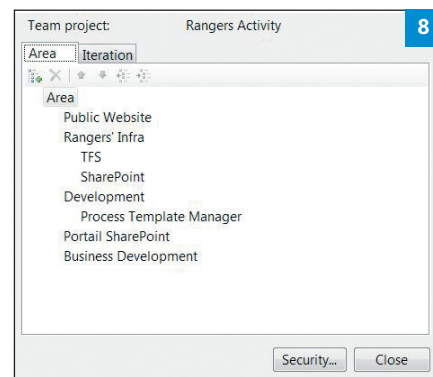
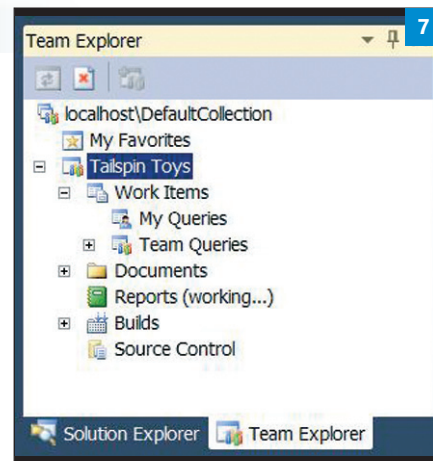
Une fois ces éléments saisis, TFS pourra ensuite enregistrer systématiquement ce qui se passe en association avec ce que vous avez renseigné ci-dessus, ce qui vous permettra de suivre l'avancement de vos projets de façon efficace.

Une filière ALM efficace passe par un pilotage aligné sur vos indicateurs de suivi. Cette étape est donc à ne pas négliger : prenez le temps de définir ce qui vous sera utile.

D'une façon générale, Il est recommandé de limiter et bien maîtriser le nombre de ces projets « Team Project » de façon à simplifier l'administration et le suivi de vos futurs projets de développement.

Toujours depuis ce menu contextuel, validez depuis l'option « Source control » les éléments suivants :

- Les check-out settings (si vous souhaitez maintenir par exemple le multiple check out)
- Les check-in Notes et Policy : cette étape de check-in est importante dans un projet, car c'est l'instant où un développeur va insérer dans votre référentiel le code sur lequel il a travaillé. Ce code va ensuite passer par différentes étapes dans votre processus d'intégration, il est donc certainement utile pour vous d'insérer à cet instant critique des points de contrôle. Nous vous recomman-



dons à minima d'ajouter l'association d'un work item à vos check-in de façon à bien piloter vos projets par la suite. Vous pouvez enrichir cette étape : **fig : 9**.

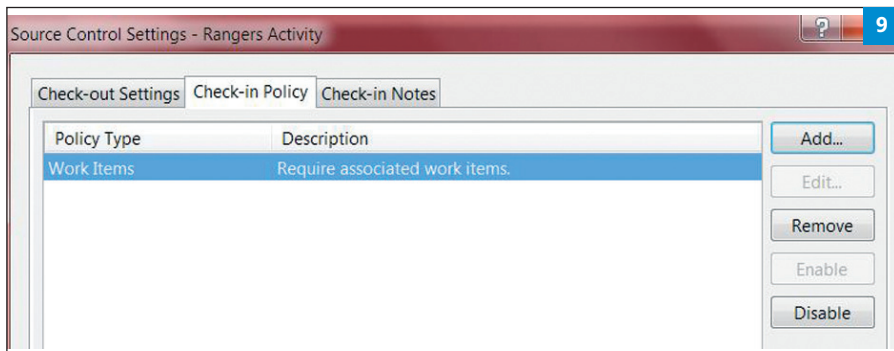
Vous disposez maintenant d'un environnement de travail sur votre serveur TFS, contenant différents dossiers, tous visibles à l'identique par les membres de l'équipe projet.

- Les **works items** qui définissent et permettent de tracer toutes les actions en cours (tâches, corrections de bugs, cas d'usage, etc.) et donc d'alimenter en informations les tableaux de bord de suivi des projets.
- La documentation (**Documents**) associée (méthodologie par exemple, usage des tests, etc.)
- Les différents rapports et tableaux de bord (**Reports**)
- L'accès au moteur de **build** ainsi qu'au module de **contrôle de source**.

Ces éléments sont aussi accessibles avec les outils préférés de chacun des acteurs du projet : Excel, Project, via un navigateur, etc.

Votre socle ALM est maintenant disponible, vous pouvez créer vos premiers projets de développement.

■ François **MERAND**  
francois.merand@sogeti.com  
Microsoft Alliance - National Practice Leader  
<http://www.dotnetrangers.net>



# Le besoin d'intégration continue

Un des principaux objectifs de l'industrialisation des projets de développement est d'améliorer la qualité des solutions produites par nos équipes. Dans le cas de projets pouvant durer plusieurs mois, il paraît évident que vérifier cette qualité en fin de chaîne, une fois tout le travail effectué, est un risque important. Que faire si rien ne fonctionne ? Sommes-nous certains que les différentes briques s'intègrent correctement les unes aux autres ? Plus simplement: est-ce que le code contenu dans le référentiel du contrôle de code source compile ? Est-ce que tous les tests unitaires sont validés ?

Il est facile d'admettre que découvrir ce genre de problèmes en fin de cycle, alors que l'on approche de la fin du projet, engendre dans la majorité des cas :

- de la panique
- des corrections rapides et parfois hasardeuses
- un décalage de la livraison du projet
- un échec complet dans certains cas plus rares.

Et tout ceci induit une qualité moyenne, voire médiocre du résultat produit.

Ne nous laissons pas abattre, il existe des solutions ! La principale d'entre elles s'appelle l'intégration continue. Il s'agit d'une pratique qui vise à inciter les développeurs à partager le plus fréquemment possible leur code, au minimum une fois par jour. Dans le cadre de Team Foundation Server, ce partage s'effectue bien évidemment à travers le contrôle de source. En plus de cela, le concept d'intégration continue nécessite une vérification d'un certain nombre de critères de qualité du travail partagé par le développeur. Cette opération est automatisée pour optimiser le travail de l'équipe à l'aide de ce que l'on appelle couramment un système de « build ». Au sein de la gamme de produits Visual Studio 2010 ALM, celui-ci est appelé « Team Build ». Ce système d'intégration continue doit être capable d'effectuer un ensemble d'opérations permettant d'obtenir un indicateur clair de la qualité du code actuellement présent dans le contrôle de source. Les opérations les plus courantes réalisées lors de ce type de processus sont :

- la compilation de la dernière version du code
- l'exécution d'outils de qualité (analyse statique de code, par exemple)
- l'exécution de tests unitaires
- l'exécution de tests de non régression fonctionnelle automatisés
- l'obtention de la quantité de code couvert par les tests exécutés

Le résultat de ces différentes actions donne lieu à la construction d'un rapport permettant de savoir où l'on en est. Si tout est une réussite, on a de bonnes raisons d'être confiants.

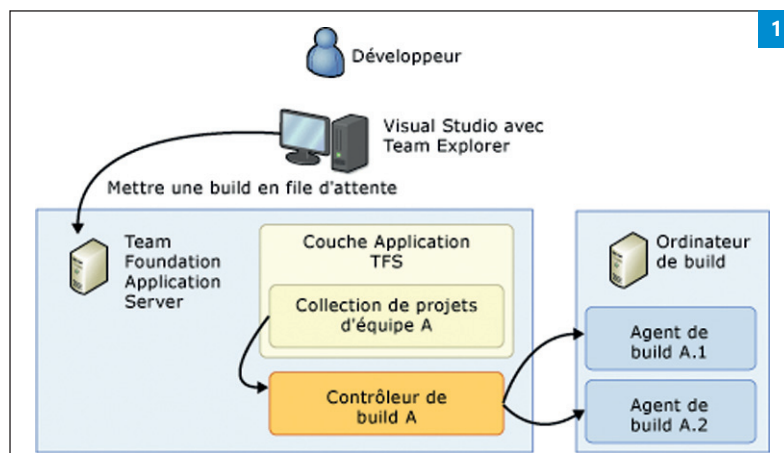
Si quelques indicateurs sont au rouge, il faut comprendre pourquoi et redresser la barre dans la bonne direction. Si au contraire tout est au rouge et que l'application ne compile pas, toute l'équipe doit se sentir concernée et effectuer les actions nécessaires à la correction des problèmes identifiés. Au final, ce processus permet de vérifier la qualité tôt et souvent. Il est en général beaucoup plus facile de corriger les problèmes au fil de

l'eau que d'un seul coup en fin de projet. Optionnellement, ces processus permettent d'effectuer des opérations automatiques supplémentaires telles que la livraison automatique de l'application au sein d'un environnement d'intégration. Ceci permet à l'équipe de test d'avoir une version toujours à jour de l'application et implicitement de détecter les anomalies rapidement.

## LE FONCTIONNEMENT DE TEAM BUILD

### Architecture des builds

Team Build est un composant de Team Foundation Server (TFS). Il s'agit d'un automate permettant d'effectuer un ensemble d'actions de manière séquentielle (le plus généralement). Ces actions sont définies dans un **flux de travail**, représenté à l'aide de la brique du framework .NET : Workflow Foundation 4.0. Les flux de travail sont exécutés par un **agent de build**, lui-même piloté par un **contrôleur de build**. L'exécution d'un



processus de build est effectué par une action d'un utilisateur (le plus souvent un développeur). Le serveur TFS est contacté et met l'exécution en file d'attente au niveau du contrôleur de build. Ce contrôleur, qui peut piloter plusieurs agents, délègue l'exécution du processus au premier agent disponible (i.e. qui n'est pas en train d'exécuter un autre processus). **fig : 1.**

## Notion de définition de build

Les flux de travail sont des modèles et ne peuvent pas être utilisés et exécutés tels quels. Il est nécessaire de les exploiter au sein de **définitions de build**. Il s'agit d'instances de ces flux de travail potentiellement configurées différemment. Si votre projet d'équipe TFS contient deux solutions Visual Studio, il est tout à fait possible de créer une définition de build pour chacune d'entre elles. L'exécution d'un processus d'intégration continue, revient en réalité à la mise en file d'attente de l'exécution d'une des définitions de build d'un projet d'équipe TFS. C'est au sein de ces définitions de builds que vous pouvez choisir les différentes actions à réaliser lors de l'exécution du processus :

- La version du code à utiliser
- La ou les solutions à compiler et dans quelle(s) configuration(s)
- Les tests à exécuter (si activé)
- Le nom à donner à la version générée à chaque exécution
- Et ainsi de suite...

## Quand s'exécute la build ?

Team Build propose un système de déclencheur poussé permettant de choisir finement le moment où le processus sera exécuté.

Il en existe 4 principaux :

- **Manuel** : L'exécution est toujours effectuée manuellement par un utilisateur.
- **Intégration continue** : Une exécution est lancée chaque fois qu'un développeur archive du code dans le contrôle de source.
- **Archivage contrôlé** (Gated Checkin) : Le code que le développeur archive est placé dans un espace privé. La build est alors exécutée en prenant la dernière version de tous les fichiers sauf ceux modifiés par le développeur. Le code n'est réellement archivé que si la build

**2**

TailspinSpyworksDEV\_20101110.2 - Archivage appliqué - <Aucune qualité assignée...>  
Afficher le résumé | Afficher le journal | Ouvrir le dossier cible | Conserver indéfiniment | Supprimer la build

Validation de l'archivage contrôlé par dpotay en tant que Ensemble de modifications 54 dans TailspinSpyworksDEV  
Exécution pour 34 secondes (Default Controller ), fin il y a 59 jours

Activité la plus récente

Dernière modification, il y a tfsbuildservice 59 jours.

Résumé

Debug | Any CPU

0 erreur(s), 0 avertissement(s)

► 1 projets/solutions compilés

► 1 série(s) de tests terminée(s) - 100 % de taux de réussite moyen (100 % de taux de réussite total)

Aucun résultat de couverture du code

est un succès. Sinon la personne est avertie qu'il reste du travail pour que les modifications soient acceptées.

- **Planifié** : Pour certaines équipes, l'intégration lors de chaque archivage peut être inutile. Parfois, une exécution par jour, la nuit par exemple, peut être suffisante. Cette option permet d'effectuer cela.

## Rapport et résultat de build

Chaque exécution de build donne lieu à un **numéro**. Celui-ci permet d'identifier la version intermédiaire créée lors de l'exécution. Lors de chaque build, le numéro est utilisé pour **appliquer une étiquette** dans le contrôle de source. Ceci permet par la suite de pouvoir identifier et télécharger la version exacte qui a été utilisée par le système pour compiler cette version intermédiaire même si le code a été modifié depuis. Le numéro de build est également utilisé pour nommer un **répertoire partagé** dans lequel le processus a déposé le résultat et les logs de la compilation ainsi que les résultats de tests.

Le rapport contient deux sections principales :

- Le **Résumé** contient les informations les plus importantes telles que le résultat de compilation et de l'exécution des tests, la liste des éléments de travail (tâche, bogues) pris en compte depuis la dernière build ainsi que celle des modifications dans le contrôle de source.
- Le **Log** permet de visualiser l'ensemble des étapes réalisées lors de l'exécution. Il s'agit concrètement de l'enchaînement des activités du flux de travail.

## Qualité de build

Le rapport d'une build propose une notion de qualité de build. Il est possible de définir une liste d'états que l'équipe de qualité peut donner à une version intermédiaire. Le processus de validation que l'on souhaite en général appliquer est de tester les versions intermédiaires jusqu'à ce qu'une de ces versions soit validée comme bonne pour livraison. Pour cela, l'équipe de qualité teste fonctionnellement la version générée et celle-ci passe à travers différents états (« premiers tests ok », « tests fonctionnels passés », etc.). Les builds marquées comme « bonne pour livraison » sont autant de versions potentiellement livrables au client.

## Aller plus loin

Par défaut, Team Build propose un flux de travail effectuant les opérations les plus couramment utilisées. Il est tout à fait possible de modifier ce flux, voire d'en créer un nouveau qui correspondrait plus à votre manière de fonctionner. Pour cela il est possible d'éditer le flux de travail qui est stocké au niveau du contrôle de code source, et de l'enrichir avec des opérations supplémentaires, appelées activités. Si l'activité en question n'existe pas, il est même possible d'en créer soi-même.

Cette souplesse dans le système de build proposé par Team Foundation Server permet d'automatiser la quasi-totalité des opérations redondantes et d'éviter la perte de temps lors du développement du projet.

■ Etienne **Margraff**  
Access It IDF



# Qu'est-ce que Team Explorer Everywhere ?

Team Foundation Server 2010 est la plateforme d'industrialisation et de travail collaboratif de Microsoft. De ce fait, certaines équipes pensent à tort que ceci est réduit à une utilisation pour des projets autour des technologies .NET ou tout du moins à une technologie Microsoft.

Depuis plusieurs années le produit TeamPrise permet de connecter l'environnement de développement Eclipse à Team Foundation Server (TFS). Il existe pour la version 2005 et 2008 et a été très bien accueilli par les nombreuses équipes l'ayant mis en place. Pour la version 2010 de TFS, ce plugin a été intégré à la gamme de produits et est maintenant édité par Microsoft sous le nom de **Team Explorer Everywhere** **fig : 1**.

Lorsque l'on regarde le team explorer proposé dans Eclipse, on retrouve exactement la même chose que dans Visual Studio 2010. Les mêmes fonctionnalités, que ce soit le contrôle de source, les éléments de travail, les rapports, ou encore les documents. Au-delà de ces fonctionnalités, il faut également mettre en avant le fait que le mot « Everywhere » n'est pas là pour rien : le plugin fonctionne dans Eclipse pour Windows évidemment, mais également pour Mac OS X, Linux, Unix, Solaris, HP-UX et IBM AIX !

## Pourquoi utiliser cet outil ?

Lors d'une réflexion globale autour de l'industrialisation dans une société, il paraît évident de s'orienter au maximum

vers un outil unique. Même si cela n'est pas toujours possible, réduire le nombre d'outils permet de diminuer la complexité et la duplication des données ou des processus.

Il n'est pas rare, au sein d'une même entreprise, de trouver plusieurs types de technologies différentes. Très fréquemment on retrouve du développement Java en parallèle des travaux réalisés avec .NET, ou du PHP combiné à des services web .NET, etc.

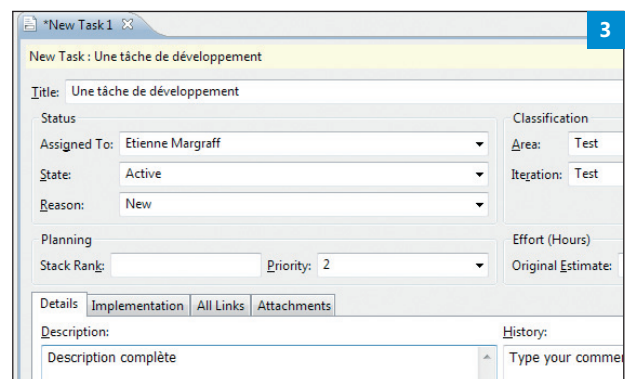
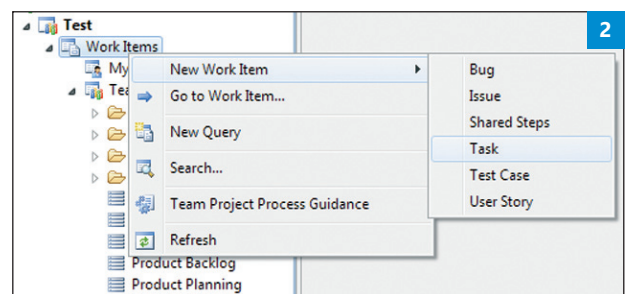
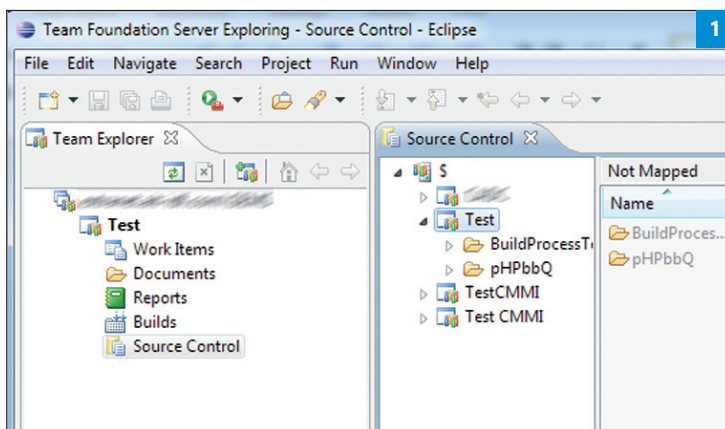
Team Foundation Server est l'outil idéal pour intégrer ces différentes technologies au sein d'un processus commun et d'une stratégie globale de gestion de projet. Grâce à l'outil Team Explorer Everywhere, il est possible d'intégrer un projet dans n'importe quel langage pour peu qu'il puisse être développé sous Eclipse. C'est le cas par exemple de Java ou de PHP (avec le plugin PHP Developer Tools), mais potentiellement avec d'autres langages. Le fait de ne pas être limité seulement à Windows ouvre des portes jusqu'à lors fermées et permet de centraliser réellement simplement l'ensemble du travail des équipes.

Les possibilités offertes par Team Foundation Server 2010 en termes de gestion de projet, de suivi d'activité et de traçabilité peuvent être alors généralisées à l'ensemble des projets. La gestion des spécifications, des tâches et des anomalies est centralisée et normalisée. De la même manière il est possible de créer des rapports qui donneront une vision globale de l'ensemble des développements, peu importe la technologie.

## LES FONCTIONNALITÉS EN DÉTAIL

### Suivi de projet

L'outil principal de suivi de projet au sein de l'outil Team Foundation Server est le système des **éléments de travail**. Il s'agit de fiches d'informations qui représentent des éléments qui ne sont ni du code, ni de la documentation. Ces éléments de travail peuvent être de plusieurs types :



on retrouve principalement des tâches, des bogues, des récits utilisateurs ou des exigences pour la gestion de spécifications ou encore des cas de test pour les aspects de validation fonctionnelle.

L'avantage de l'outil est qu'il est égal en tout point à son équivalent dans Visual Studio.

Les utilisateurs habitués à celui-ci ne sont pas perdus et les repères reviennent rapidement. Un simple clic droit permet de créer un élément de travail de la même manière, par exemple : **fig : 2**.

L'interface de saisie est également identique. Tout comme dans la version web, les modifications apportées sur le formulaire sont répliquées dynamiquement sur la version que l'on visualise dans Eclipse. Par exemple, si l'équipe a décidé d'ajouter un champ contenant le responsable technique sur une tâche de développement, celui-ci apparaîtra ici

également : **fig : 3**. De la même manière, il est possible de créer, visualiser et modifier les requêtes permettant de visualiser une liste d'éléments de travail. Il est très facile de créer, par exemple, une requête donnant la « Todo list » d'un développeur : **fig : 4**.

Bien évidemment, les requêtes donnant un résultat hiérarchique et les liens entre les éléments sont gérés. Ceci permet d'avoir accès à toute la traçabilité offerte par l'outil, même depuis Eclipse !

## Utilisation du contrôle de code source

Le contrôle de source est sans doute un des éléments principaux d'un outil de collaboration pour une équipe de développement. Il doit être le plus complet possible en termes de fonctionnalités et celles-ci doivent être les mêmes pour n'importe quel utilisateur technique.

Team Explorer Everywhere offre les mêmes fonctionnalités que Visual Studio au niveau de la gestion des opérations possibles avec le référentiel de sources.

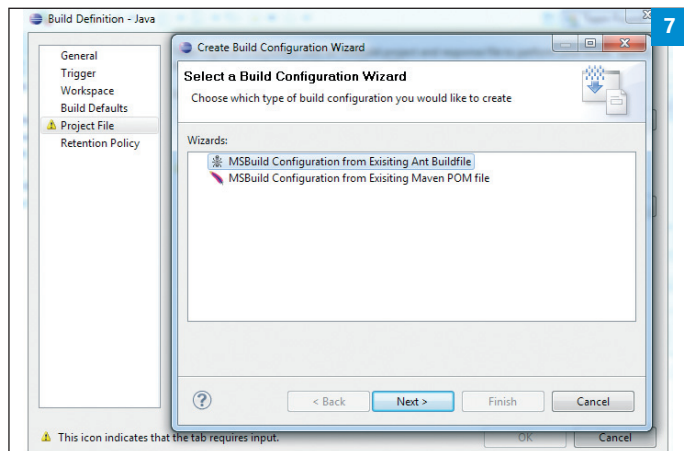
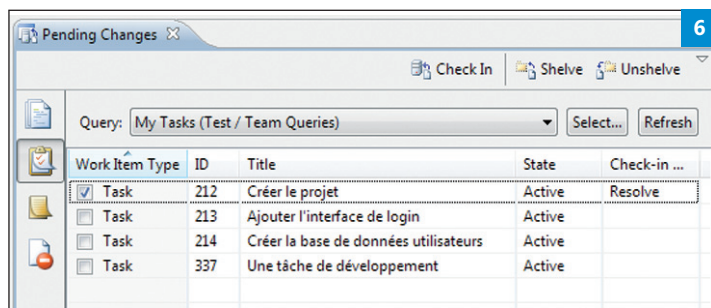
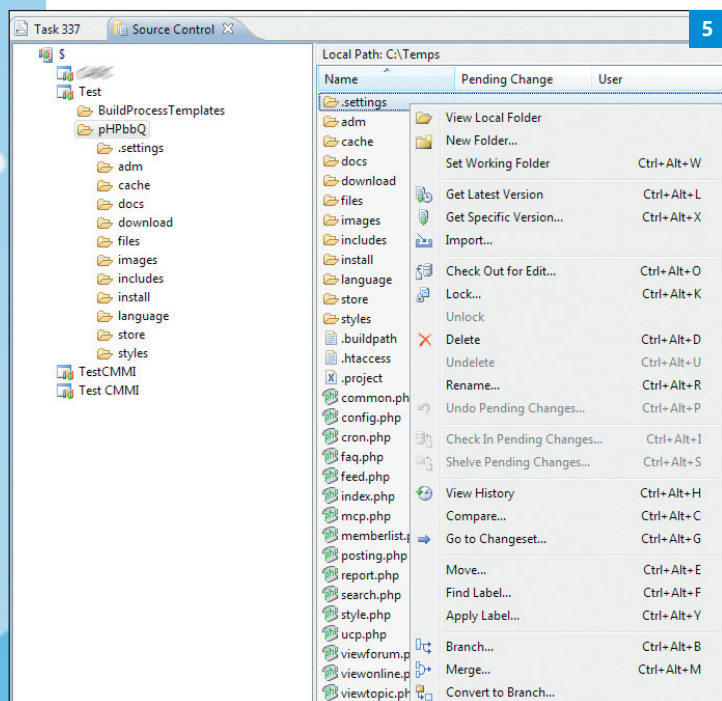
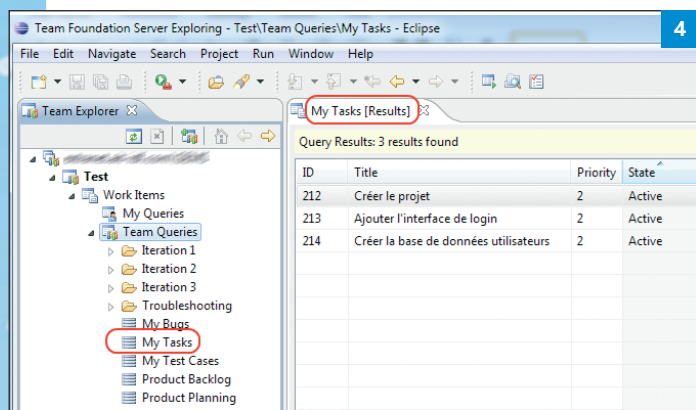
Les opérations les plus standard sont bien évidemment disponibles, mais pas uniquement. Comme pour un projet Visual Studio, il est possible d'identi-

fier une version avec une étiquette, de mettre en place le modèle promotionnel, de créer des branches de correction de bogues et de reporter les modifications d'une version à une autre **fig : 5**.

En termes de traçabilité, il est possible d'associer une tâche ou un bogue lors de l'archivage de modifications (checkin). Ceci permet de créer un lien entre cette tâche et les modifications apportées au code. Par la suite cela permettra de connaître le travail qui a été réalisé par le passé. Si un bogue réapparaît on se pose souvent la question : « nous l'avons corrigé, pourquoi est-ce qu'il revient ? ». La lecture du code écrit à l'époque peut être très intéressante **fig : 6**.

## Gestion de l'intégration continue

L'intégration continue permet d'automatiser les processus de validation de la qualité du code produit. La plateforme Team Foundation Server permet de réaliser ceci via le système appelé Team Build. Lors de développement .NET, celui-ci permet d'automatiser la compilation de l'application, l'exécution des tests et d'outils d'analyse, et de fournir un rapport détaillé. Pour le cas de Java, il est possible d'avoir exactement la même chose grâce au Team Explorer Everywhere. Lors de la création d'une définition



(script) de build, l'outil demande un script Ant ou Maven 2. Tout le reste est automatiquement géré par l'outil qui se chargera de le planifier et l'exécuter **fig : 7**.

Il est alors possible de l'exécuter manuellement comme on le ferait via le team explorer classique. Le rapport en temps réel est légèrement différent, mais l'information que l'on obtient en fin de processus est identique : **fig : 8**.

Il est également intéressant de noter que le nouveau système dit d'archivage contrôlé (Gated Checkin) est également disponible au sein d'Eclipse. Cette fonctionnalité permet de lancer l'exécution d'une build d'intégration continue lors d'un archivage. La différence avec une intégration continue classique est que l'archivage est réalisé uniquement si la build est un succès. Sinon, le développeur est averti qu'il faut effectuer une opération de correction pour que son code réponde aux critères de qualité du projet.

*Note : Le développeur n'est pas bloqué en attendant la fin du processus de vérification, il peut continuer à travailler et pourra resynchroniser son espace de travail local à la fin de celui-ci.*

## Utilisation des rapports

L'avantage d'avoir toutes les informations du projet stocké au même endroit est avant tout qu'il est possible de réaliser facilement une consolidation de l'information. Avec Team Foundation Server 2010, un ensemble de modèles de rapports sont disponibles en standard et permettent de suivre l'avancement du projet, l'évolution de la qualité du code, la vitesse de prise en compte des anomalies ou encore la quantité de travail réalisé pour la validation fonctionnelle de l'application. Team Explorer Everywhere

permet de visualiser la liste de ces rapports et de les afficher directement dans Eclipse : **fig : 9**.

## Utilisation du portail documentaire

Un projet informatique contient un certain nombre de documents qu'il est nécessaire de conserver, de faire évoluer. Très souvent d'ailleurs, on aura plusieurs versions pour le même document. Team Foundation Server 2010 est connecté à un site Sharepoint qui permet de remplir principalement ce rôle. Les documents sont stockés dans ce portail qui permet également d'avoir accès à un tableau de bord de synthèse du projet. Au-delà de cela, Sharepoint offre de nombreuses fonctionnalités qu'il est possible de mettre en place telles qu'un Wiki pour l'équipe, un calendrier commun pour le projet et bien d'autres **fig : 10**.

Les documents peuvent être accédés via le site mais également à partir d'Eclipse directement. Ceci permet de changer moins souvent d'outils et de rester au maximum dans l'environnement de développement pour optimiser le travail : **fig : 11**.

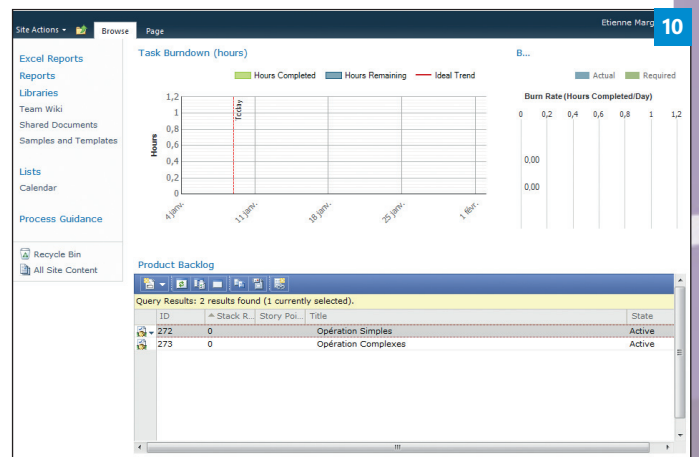
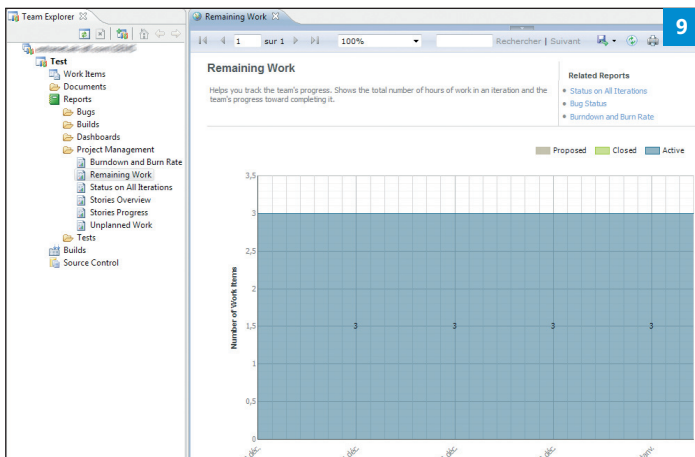
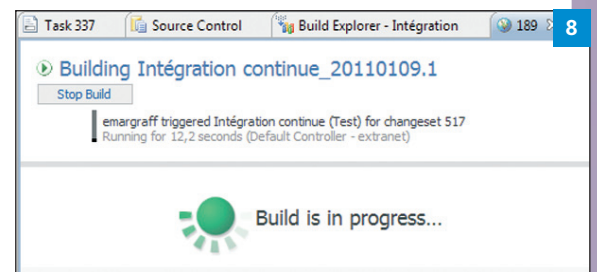
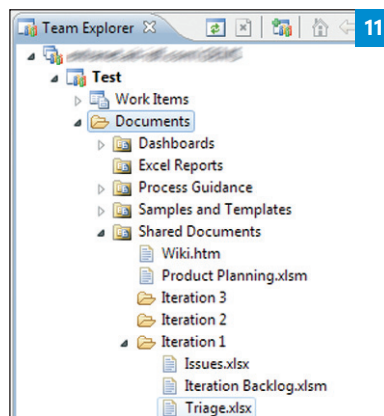
## Qu'en est-il des autres environnements de développement ?

L'intégration de Team Foundation Server 2010 au sein d'Eclipse est un des exemples les plus complets d'extensibilité de la plateforme mais ce n'est pas le seul cas. Par exemple, en ce qui concerne le contrôle de source et la possibilité de communiquer avec le serveur, c'est possible pour les outils suivants :

- Visual Studio .NET 2003
- Visual C++ 6 SP6
- Visual Visual Basic 6 SP6
- Visual FoxPro 9 SP2
- Microsoft Access 2007
- SQL Server Management Studio
- Enterprise Architect 7.5
- PowerBuilder 11.5
- Microsoft eMbedded VC++ 4.0

Ceci est possible grâce aux providers MSSCII (qu'il est possible de télécharger sur le site Visual Studio Gallery, même si l'intégration est moins complète et se limite au code source, l'intérêt reste entier. Il suffit d'utiliser le team explorer en parallèle de l'environnement de développement pour avoir accès aux autres fonctionnalités.

■ Etienne Margraff  
Access It IDF





# Mise en œuvre d'une galerie privée d'extensions dans Visual Studio

Visual Studio 2010 a apporté son lot d'améliorations au niveau de l'extensibilité en proposant un nouveau modèle d'extension téléchargeable à partir d'une galerie publique. Cette galerie accessible à partir du gestionnaire d'extensions ne permet pas de publier des extensions privées à votre entreprise. Il existe cependant un moyen de le faire en créant une nouvelle extension rajoutant un onglet dans le gestionnaire d'extensions. Nous verrons dans cet article les mécanismes à utiliser pour y arriver.

Le gestionnaire d'extension accessible à partir du menu « Outils » se matérialise sous la forme d'une boîte de dialogue munie de plusieurs onglets. Le premier donne accès à une liste d'extensions téléchargeables à partir de la galerie de Microsoft, le second permet de lister les extensions déjà installées avec la possibilité de les désactiver ou de les supprimer, enfin le dernier affiche les mises à jour disponibles **Fig 1**. Chaque extension correspond à un fichier VSIX permettant de déployer une extension une fois téléchargée. Un simple fichier VSIX contient les éléments suivants :

[Content_Types].xml	Requis par le format OPC pour reconnaître le contenu du fichier
extension.vsixmanifest	Fichier contenant les données (metadata) décrivant l'extension.
Fichiers de l'extension	Fichiers nécessaires au fonctionnement de l'extension (binaires ou autres)
Fichiers associés	Ce peut-être les images, icônes, fichiers texte de licence...

Il est à noter que contrairement aux anciennes versions de Visual Studio, il n'est plus nécessaire de disposer des droits d'administrateur pour installer une extension. Elles sont copiées dans le répertoire de l'utilisateur courant et rien n'est mis dans le GAC.

Un fichier VSIX peut contenir une ou plusieurs extensions. Les extensions supportées par ce mode de déploiement sont les suivantes :

- **Template de projet/Item** : Une liste de templates Visual Studio qui seront accessibles dans la boîte de dialogue affichée lors de la création d'un nouveau projet/item.
- **Wizard de Template** : Une librairie (associée à un ou plusieurs templates) contenant un assistant qui affichera une interface de saisie ou exécutera du code lors de l'exécution d'un template.
- **Package Visual Studio** : Un package, qui correspond à un module d'extension qui s'intégrera dans Visual Studio, avec son fichier pkgdef associé

• **Composant MEF** : Une librairie qui sera trouvée et consommée par le gestionnaire d'extension de Visual Studio pour enrichir une extension existante. C'est ce type de composant que nous allons utiliser pour ajouter notre galerie d'extension privée.

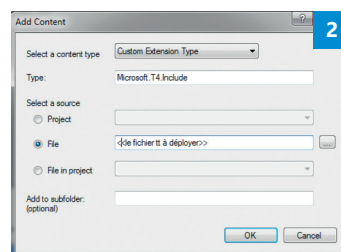
• **Contrôle de boîte à outils** : Une librairie contenant un ou plusieurs contrôles s'affichant dans la boîte à outils de Visual Studio pour être utilisé dans les différents designers graphiques.

Les extensions suivantes ne sont pas déployables avec un fichier VSIX : Macros, add-ins, code snippet. On pourra, dans ce cas, utiliser un fichier msi.

## Les extensions personnalisées

Le fichier VSIX permet aussi de déployer des extensions personnalisées qui ne correspondent pas à celles décrites ci-dessus. Ceci est utile, par exemple, pour pouvoir déployer des artefacts à vos propres extensions. Un cas concret de l'utilisation de ce type d'extension est fourni par le moteur de génération de code de Visual Studio (T4) qui permet de déployer vos propres templates de génération (.tt) par ce moyen.

Pour cela, vous devez créer un nouveau projet de type VSIX dans Visual



Studio et configurer le fichier manifest en lui précisant que vous souhaitez déployer une extension personnalisée de type « Microsoft.T4.Include » **fig. 2**.

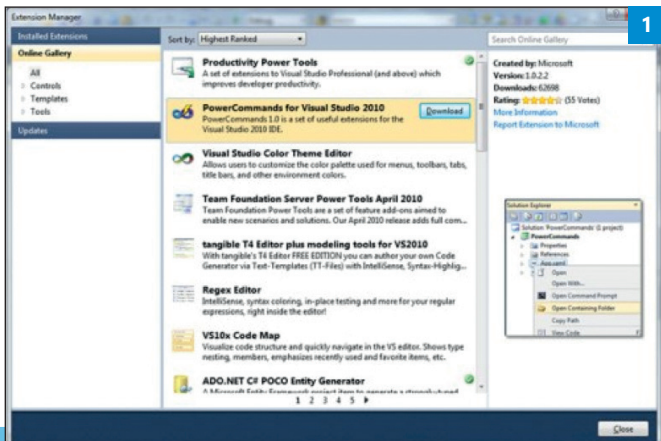
Vos templates seront alors automatiquement détectés dans Visual Studio et pourront

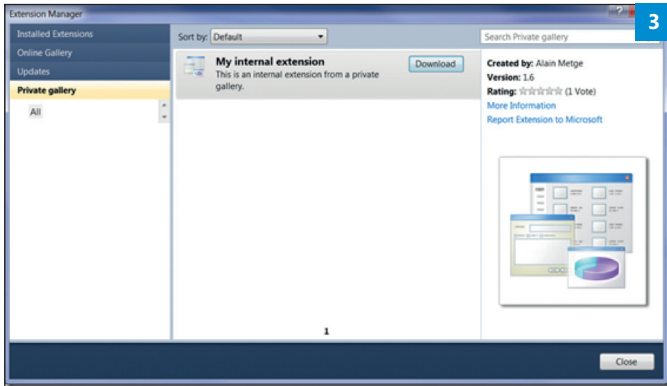
être utilisés très simplement avec un fichier tt contenant une simple directive **@include** du style : `<#@ include file="monTemplate.tt" #>`

## Mise en œuvre de la galerie privée

Tout le mécanisme d'extensions de Visual Studio utilise MEF pour son fonctionnement interne. Le gestionnaire d'extensions ne déroge pas à la règle. Lors de son chargement, il va d'abord initialiser les onglets (les providers) avec 3 providers de base : Un provider pour les extensions installées localement sur le poste, un autre pour les extensions en ligne (qui interroge la galerie publique Visual Studio) et un dernier pour gérer les mises à jour. Dans notre cas, la mise à disposition de notre galerie privée va consister à ajouter un provider personnalisé donnant accès à nos propres extensions **fig. 3**.

Pour l'activer dans le gestionnaire d'extension, il nous faut créer une extension implémentant l'interface **Microsoft.VisualStudio.Extens-**





**sionManager.IVsExtensionManagerDialogProvider** et rajouter l'attribut **MEF Export** avec ce même type en paramètre.

Le type à hériter se trouve dans la DLL `Microsoft.VisualStudio.ExtensionsManager` que vous trouverez dans le répertoire `...\Microsoft Visual Studio 10.0\Common7\IDE\PrivateAssemblies`. Ce qui nous donne :

```
1: [Export(typeof(IVsExtensionManagerDialogProvider))]
2: class PrivateExtensionManager : IVsExtensionManagerDialogProvider
3: {
```

Une fois, la classe implémentée, vous n'avez plus qu'à créer un package d'extension et publier cette extension en tant que MEF Extension....

**Nota :** Le fonctionnement décrit ici utilise un mécanisme qui n'est pas supporté officiellement par Microsoft du fait de l'utilisation d'une librairie contenue dans le répertoire `PrivateAssemblies`. Vous devrez vous assurer de son bon fonctionnement lors de chaque changement de version de Visual Studio.

L'interface **IVsExtensionManagerDialogProvider** hérite elle-même de l'interface **IVsExtensionsProvider** qu'il va falloir implémenter.

```
public interface IVsExtensionsProvider
{
    object DetailViewDataTemplate { get; }
    IVsExtensionsTreeNode ExtensionsTree { get; }
    object LargeIconDataTemplate { get; }
    object MediumIconDataTemplate { get; }
    string Name { get; }
    object SmallIconDataTemplate { get; }
    float SortOrder { get; }
    IVsExtensionsTreeNode Search(string searchTerms);
}
```

La manière la plus simple pour l'implémentation est d'utiliser comme classe de base la classe `Microsoft.VisualStudio.ExtensionsExplorer.UI.VsExtensionsProvider`.

Les principales méthodes à implémenter sont :

- **DetailViewDataTemplate** et **MediumIconDataTemplate** correspondent aux DataTemplate XAML qui serviront à l'affichage. Ici, puisque nous voulons nous insérer dans le gestionnaire de Visual Studio, le mieux est d'utiliser les templates existants. Ce qui peut se faire de la manière suivante :

```
private object _detailViewDataTemplate;
public override object DetailViewDataTemplate {
    get {
        if (this._detailViewDataTemplate == null) {
            ResourceDictionary dictionary = new ResourceDictionary();
            dictionary.Source = new Uri("Microsoft.VisualStudio.ExtensionsManager.Implementation;component/UI/OnlineProviderTemplates.xaml",
```

```
UriKind.Relative);
            this._detailViewDataTemplate = dictionary["OnlineDetailTemplate"];
        }
        return this._detailViewDataTemplate;
    }
}
private object _mediumIconDataTemplate;
public override object MediumIconDataTemplate
{
    get {
        if (this._mediumIconDataTemplate == null) {
            ResourceDictionary dictionary = new ResourceDictionary();
            dictionary.Source = new Uri("Microsoft.VisualStudio.ExtensionsManager.Implementation;component/UI/OnlineProviderTemplates.xaml",
            UriKind.Relative);
            this._mediumIconDataTemplate = dictionary["OnlineTileTemplate"];
        }
        return this._mediumIconDataTemplate;
    }
}
```

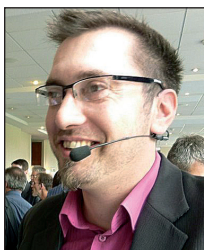
- **ExtensionsTree** fournit la liste des extensions à afficher. Pour être compatible avec les dataTemplates précédemment citées, le modèle de données doit avoir les propriétés suivantes :

IsSelected	Indique si l'extension est sélectionnée
Name	Nom de l'extension
VsixID	Identifiant unique de l'extension
Description	Texte descriptif
DownloadUrl	Url de téléchargement
Author	Nom du ou des auteurs
VsixVersion	Version de l'extension
DownloadCount	Nombre de téléchargements
Icon	Url de l'icône à afficher
Rating	Nombre d'étoiles
RatingsCount	Nombre de votes
MoreInfoURL	Url sur un site d'aide

Cette méthode doit renvoyer un objet implémentant **IVsExtensionsTreeNode** dont la propriété **Extensions** est remplie avec la liste des extensions (sous la forme d'instance de **IVsExtension**)

Comme nous le voyons, tout ceci n'est pas forcément trivial car cela nécessite d'implémenter un certain nombre d'interfaces de Visual Studio. Heureusement, la publication du projet NuGet (<http://nuget.codeplex.com>) par Microsoft permet d'avoir accès à un exemple d'implémentation qu'il est facile de réutiliser. Vous pouvez aussi trouver un article plus détaillé sur la mise œuvre d'une telle solution sur le site [www.dotnetrangers.net](http://www.dotnetrangers.net) qui reprend point par point toutes les étapes précédemment décrites. La dernière étape consistera à mettre en place le serveur hébergeant la galerie privée. Le gestionnaire d'extensions utilise le protocole OData pour le solliciter, ce qui peut se coder facilement avec WCF. De la même manière que le code de l'extension Visual Studio, vous trouverez un exemple d'implémentation sur le site des dotnetRangers. Nous avons vu brièvement comment étendre Visual Studio pour lui rajouter une galerie privée. Ce qui est un vrai manque dans la version actuelle. Gageons cependant que cette fonctionnalité sera rajoutée par Microsoft dans les futures versions. Dans tous les cas, cela nous a permis de voir la puissance du mécanisme d'extension mis en place dans la version 2010.

■ Alain Metge - dotNetRangers by Sogeti



## Noham Choulant

### « un projet commence par l'expression des besoins »

Noham travaille en environnement Microsoft depuis plusieurs années. Expert et MVP ALM, il est en poste à Evaluant, SSII spécialisée dans les technologies Microsoft. Noham revient pour nous sur son expérience TFS. Son site : <http://choulant.blogspot.com>

*Y-a-t-il un avant et un après Team Foundation Server ?*

**Noham :** Un projet commence par une expression des besoins, par la rédaction des spécifications fonctionnelles. Bien que l'on parle beaucoup de méthodologie agile aujourd'hui il ne faut pas en oublier l'essentiel qui est la réussite du projet. La mise en place d'une méthode ne veut pas dire zéro document, donc avant tout développement il faut prévoir, planifier, piloter le projet. Dans des sociétés comme les SSII, une partie de ce travail est faite par les clients, c'est pour cela que l'on peut trouver l'utilisation de Team Foundation Server assez tôt dans le cycle de développement. Dans des sociétés d'une certaine taille où la DSI n'est qu'une petite partie de l'entreprise, l'avant TFS est bien plus important, et passe notamment par la sélection de projet (Portfolio manager).

*La mise en place d'un ALM modifie souvent l'organisation du travail et le formalisme. Comment avez-vous abordé ces changements ?*

**Noham :** Chaque société a sa propre gestion du changement, j'ai vu certaines d'entre-elles étudier la mise en place des outils pour l'ALM avec une préparation de marathonien (étude, réflexion, analyse, vérification, entraînement) d'autres au contraire foncent tel un aigle sur sa proie et réfléchissent ensuite à « quoi faire » et « comment le

faire ». Il n'y a pas une façon de faire meilleure que l'autre. Le changement est important dans une équipe de développement, les notions de gestion et de pilotage ne sont pas pour tous des idées et des actions forcément claires. Je pense que certaines sociétés ne doivent pas minimiser cette étape et doivent se faire accompagner. Dans mes équipes nous avons pris le temps de mettre en place ces nouvelles notions et nous en apprenons tous les jours.

*TFS 2010 introduit de nombreux changements, notamment dans la planification ou encore la modélisation projet. Avez-vous changé vos méthodes de définition d'architecture et de suivi projet ?*

**Noham :** En tant que chef de projet, c'est plus tôt sur le suivi de projet que les changements sont les plus significatifs. Avant nous utilisions TFS 2005 et 2008 plus comme un contrôleur de code source que comme une plateforme d'industrialisation logicielle. Intégrer l'ensemble des briques de cette nouvelle version comme la validation de l'architecture, les tests fonctionnels la hiérarchisation des éléments de travail sont autant de nouvelles fonctionnalités qui m'ont permis de gagner en pilotage et ainsi de mieux anticiper les changements.

*Un ALM doit permettre une meilleure communication entre les équipes, entre la technique et le métier. Comment ressentez-vous cela au quotidien ?*

**Noham :** Cette communication n'a jamais été simple, le métier s'intéressant d'abord à l'utilisation du produit et la technique, aux capacités de réalisation. Je n'irai pas jusqu'à dire que TFS permet de mieux communiquer mais il permet de mettre en place des outils et des méthodes pour aider. Des outils pour Excel avec l'add-in de Microsoft) ou

Word (avec TeamForWord) ont permis à nos métiers d'exprimer le besoin, de formaliser les exigences de manière simple et collaborative avec les équipes de conception. Les rapports et les applications web (SharePoint, Team Web Access) ont permis d'utiliser les mêmes outils dans les deux mondes et d'améliorer ainsi cette collaboration.



## Dany Damaj

architecte

.net, apporte son retour d'expérience autour de l'usage de TFS.

L'adoption de TFS est rapide et il n'est pas rare de se demander "comment faisait-on avant ?". Le contrôle complet du code source, l'intégration avec Visual Studio et la liaison des "work items" avec des éléments concrets tels qu'une couche d'un diagramme de couches, sont autant de fonctionnalités qui modifient le quotidien de l'architecte, du développeur et du testeur .NET.

**Prendre en compte les modifications d'organisation et fonctionnelles.**

Ces changements ont été abordés de manière globale. Un nouveau projet a été l'occasion de mettre en place un ALM complet et d'utiliser l'ensemble des outils Microsoft (Project Server, TFS 2010, et les nouveautés de la version Ultimate de Visual Studio 2010) pour soutenir totalement les nouveaux processus. Une vraie révolution !

**Le nouveau modèle "Projet de modélisation" de Visual Studio 2010**

Ultimate a été largement employé. Plutôt que d'utiliser des outils tiers pour modéliser nos



# Pascal Laforest

Chef de projet chez Access it



Après 10 ans en tant que consultant et chef de projet BI, Pascal Laforest est depuis 2009 directeur des opérations d'Access It IdF. Dans ce cadre, il assure notamment la direction de projets, notamment au forfait, en environnement .Net et SQL Server. C'est dans ce contexte qu'il utilise depuis 2 ans maintenant environ la solution TFS, avec les équipes projets.

*Y-a-t-il un avant et un après Team Foundation Server ?*

**Pascal :** Clairement, et pour moi cela se résume en un point : le regroupement de l'ensemble des outils existants dans un référentiel unique. Avant TFS, j'utilisais plusieurs outils, un bogue tracker dans un coin, un

diagrammes UML, ceux-ci ont été directement réalisés avec Visual Studio et liés à des "work items" de TFS. Il en va de même pour les diagrammes de couches, qu'il s'agisse des diagrammes d'architecture fonctionnelle, logique ou physique. Ces diagrammes nous ont permis de vérifier régulièrement la conformité du code par rapport à l'architecture définie au départ.

## Une évolution du métier d'architecte ?

Les concepts fondamentaux restent les mêmes, notamment le processus principal permettant de partir du Cahier Des Charges, et des objectifs énoncés par le client, pour aboutir à une architecture concrète exploitée ensuite par les développeurs. La mise en oeuvre de ces concepts, en revanche, évolue quasiment à chaque nouveau projet, étant donné la vitesse d'évolution des outils. Ainsi, chaque changement de version de TFS, de Visual Studio, de Framework .NET ou encore d'Office, permet d'envisager de nouvelles améliorations. Le rôle de chaque membre de l'équipe est mieux défini et respecté. La prise de contrôle est complète, et englobe la totalité de la vie du projet. Un changement aussi important peut être mal vécu s'il est mal préparé. Un membre de l'équipe peut avoir l'impression d'être surveillé et jugé plus facilement. L'ALM ne doit pas se détourner de son objectif principal : le respect des coûts et des délais des projets.

wiki et un référentiel de documents dans un autre, quelques fichiers Excel pour la communication et un plan projet pour la gestion du projet et la consolidation des tâches... La consolidation de ces données était douloureuse et la collecte des informations nécessitait pas mal d'échanges oraux ou par e-mail avec les développeurs. Dans Team Foundation Server, un seul référentiel, une multitude de clients et donc chaque intervenant du projet met à jour ses données depuis son outil, ce qui représente un gain de temps considérable pour le chef de projet et garantit une fiabilité des données consolidées.

*La mise en place d'un ALM modifie souvent l'organisation du travail et le formalisme. Comment avez-vous abordé ces changements ?*

**Pascal :** Nous avons mis en place l'outil par étape : d'abord la gestion du contrôle de code source pour habituer les développeurs, avec peu de changement vis-à-vis des habitudes acquises dans SourceSafe.

Ensuite, nous avons migré la gestion de projet vers les éléments de travail de type tâche et la gestion des incidents au travers des fiches de bogues. Quelques semaines après, notre gestion de projet a évolué pour prendre en compte sous la forme de fiches les récits utilisateurs et les cas de tests liés. Enfin, dernière brique mise en place récemment, nous avons ajouté la prise en charge de la génération automatique de livrables directement depuis TFS.

*TFS 2010 introduit de nombreux changements, notamment dans la planification ou encore la modélisation projet. Avez-vous changé vos*

*méthodes de suivi projet ? (en particulier sur les notions de qualité logicielle et d'industrialisation logicielle)*

**Pascal :** En jouant le jeu du découpage en récits utilisateurs et en tâches techniques dans TFS, je bénéficie maintenant de rapports vraiment pertinents et intéressants pour le suivi de projet. Là où avant je consolidais des données issues de plusieurs copier / coller dans Excel, je peux maintenant interpréter et diffuser les rapports Reporting Services tels que ceux de progression des récits, d'indicateurs de qualité et de tendances des bogues. En plus de cela, je ne monopolise plus du temps de mon équipe en faisant des réunions pour savoir où on en est, qui a fait quoi ou encore combien de temps cela lui a pris, tout est agrégé en un seul point pour moi.

*Un ALM doit permettre une meilleure communication entre les équipes, entre le Technique et le Métier. Comment ressentez-vous cela au quotidien ?*

**Pascal :** L'information est moins volatile, les emails sont seulement utilisés comme systèmes d'alerte pour notifier des modifications dans le référentiel. Là aussi, le récit utilisateur joue un rôle essentiel car c'est l'artefact qui est utilisé comme base de travail et consommé par tous : le métier depuis le portail SharePoint, le chef de projet depuis MSPProject, le testeur Test Manager 2010 et le développeur dans Visual Studio ! Plus que tout, en revanche, je ne voulais pas que la communication « réelle » soit annihilée. Je reste persuadé que les échanges d'humain à humain apportent de grands bénéfices au projet. TFS permet surtout de réduire ce type d'échanges à l'essentiel.

# Les gammes : à vous de choisir !

Avec Visual Studio (VS) 2010, Microsoft a rationalisé l'ensemble de la gamme afin de la rendre plus compréhensible. D'autre part, pour répandre l'usage de l'ALM, TFS est disponible par défaut avec plusieurs éditions de Visual Studio 2010. Pour bénéficier de TFS, vous devez avoir un abonnement MSDN.

Fonctionnalités	VS 2010 Ultimate + MSDN	VS 2010 Premium + MSDN	VS 2010 Professionnel + MSDN	VS Test Professionnel + MSDN
<b>Support de la plateforme de développement</b> Windows, Web, Cloud, Office, Sharepoint	•	•	•	
<b>Architecture et modélisation</b> Explorateur d'architecture, diagrammes de couches, validation des dépendances Diagrammes en lecture seule (UML, couches, graphes DGML)	• •	•		
<b>Développement de bases de données</b> Déploiement, gestion des changements, génération de données de test, test unitaire	•	•		
<b>Debug, diagnostics, tests</b> IntelliTrace, Test de performance web, test de charge Analyse statique de code, métrique de code, profiling, couverture de code, analyse des tests impactés, tests d'UI codés Test unitaire	• • •	• •	•	
<b>Test and Lab Management</b> Test and Lab Manager : gestion des cas de tests, environnement d'exécution de tests manuels, enregistrement des tests manuels Provisionnement d'environnements virtuels à la demande, création à partir de template	• •			• •
<b>Team Foundation Server</b> Contrôle de source, gestion des éléments de travail, intégration continue, portail d'équipe, collecte et suivi d'indicateurs, guide de planning agile	•	•	•	•
Visual Studio Team Foundation Server 2010 + une licence d'accès client	•	•	•	•
Microsoft Office Professional Plus 2010, Project Professional 2010, Visio Premium 2010, Expression Studio (seul Ultimate contient Sketchflow)	•	•		
Windows Azure	•	•	•	
Windows (client et server)	•	•	•	•
SQL Server	•	•	•	•
Toolkits, Software Development Kits, Driver Development Kits	•	•	•	•
Microsoft Office	•	•		
Microsoft Dynamics	•	•		
Tous les autres serveurs	•	•		
Windows Embedded operating systems	•	•		
Collections Microsoft E-Learning	2	2	1	1
Incidents de support technique	4	4	2	2
Support prioritaire dans les forums MSDN	•	•	•	•

## Le TFS Basic

Il existe une édition légère de TFS, TFS Basic. Il se destine aux petites équipes de développement utilisant Visual SourceSafe (ou équivalent), ou aucun outil. Il s'agit d'une solution

idéale pour le contrôle de source, le suivi des bugs, la gestion des cas de tests, et l'automatisation des builds. Basic utilise le même serveur que TFS, sans limitation. Cette édition a l'avantage

d'être très simple à déployer. Vous disposez d'une console d'administration complète et en cas de besoin, TFS Basic s'étend jusqu'à disposer d'un TFS 2010 complet.

l'équivalent de cette requête en SQL. Pour comparer, le code ci-dessus a été écrit et testé en trois minutes.

## Performances

Pour continuer sur l'exemple précédent, non seulement la requête était très rapide à écrire, mais elle est également performante : 1570 ms pour traiter 250 000 objets (sur un AMD 4400+ avec 1 Go RAM standard). Et tous les caches et les index d'un SGBD ne pourront rien faire sur ce genre de traitement : la restriction complexe oblige à un Full Scan, et le barème dépendant de la date du jour n'est pas précisément le point fort des bases de données.

Les SGBD restent très performants sur de nombreuses requêtes, mais dès qu'on commence à entreprendre un fonctionnement métier un peu complexe, il est souvent nécessaire de recourir à des astuces de traitement qui rendent les requêtes difficilement maintenables. Les index améliorent bien sûr fortement les performances, mais cette technologie existe également sur les langages grâce aux dictionnaires hashés. Du coup, l'augmentation de performance par la mémoire redevient un avantage.

## LES LIMITES DE LA PRÉVALENCE

Arrivés à ce point, et si j'ai bien fait mon travail, vous êtes censés vous dire « *C'est absolument génial, pourquoi est-ce que ce n'est pas utilisé plus souvent ?* » Peut-être vous dites-vous qu'il doit y avoir un problème... Tout d'abord, la prévalence n'est pas une technologie miracle qui va sans aucun effort vous permettre d'augmenter par dix la performance de vos serveurs et de supprimer tout votre mapping O/R. Comme toute autre technologie, elle a également ses inconvénients. En fonction des projets, ceux-ci peuvent être rédhibitoires, et il vaudra mieux continuer à payer vos licences de SGBD. Ou bien ils sont sans importance par rapport aux gains entrevus, et vous tenez peut-être un bon candidat à ré-architecture sous forme prévalente. Mais listons d'abord ces inconvénients :

### Pas de support de SQL

Tout d'abord, vous pouvez mettre au placard toutes vos requêtes SQL. Elles n'ont pas d'utilité sur une application prévalente, car on ne manipule pas des tables de données, mais des ensembles non tabulaires, avec une structuration arbitraire. Evidemment, dans certains cas, cela peut avoir du sens de recréer une structure proche ou avec un mapping léger pour pouvoir réutiliser de la grammaire SQL. Une option consiste alors à créer une grammaire de requête spécifique avec un métalangage, voire à utiliser un parseur de SQL pour simuler un connecteur à vos données prévalentes.

Vous pouvez trouver un article à ce sujet sur le blog DotNet-France de l'auteur, montrant comment on peut utiliser la grammaire M (livrée dans Microsoft Oslo) pour décomposer une grammaire quelconque et créer des expressions lambda qu'on inclura dans des expressions Linq afin de requêter un contexte prévalent. Pour aller plus loin, il faudrait idéalement créer un connecteur ADO.NET vers une base prévalente, en se basant sur des conventions de liens entre les noms utilisés et la structure des objets dans le contexte prévalent. Toutefois, ceci représenterait une sorte de retour vers un mapping. La piste est actuellement explorée, mais le seul requêtage qui ait du sens en prévalence est d'utiliser Linq-To-Objects.

### Pas d'éditeur de données CRUD

La prévalence a été présentée comme architecture de mise en pratique du Domain-Driven Development (et plus particulièrement de CQRS) lors de plusieurs sessions de l'Agile Tour 2010. A chaque fois, la question d'un éditeur simple est revenue. Les fichiers de logs et de snapshots liés au moteur de prévalence sont en effet une sérialisation binaire, et non lisibles en dehors de l'application dédiée, d'où le besoin ressenti d'avoir un moyen standard de voir le contenu des données.

La première partie de la réponse consiste à souligner que le format binaire ne pose pas de problème en soi, et est absolument nécessaire pour la performance. Aucun SGBD digne de ce nom ne stockant ses données dans des fichiers lisibles par un humain, il n'y a aucune raison que la prévalence fasse différemment. Vos fichiers prévalents n'ont pas de sens en dehors de l'application liée de la même manière que vous ne pouvez pas utiliser les fichiers .mdf et .ldf de SQLServer dans Oracle.

La seconde partie de la réponse est liée à la présence de l'éditeur lui-même. Il n'existe effectivement rien de tel pour l'instant dans les frameworks de prévalence actuels (Prevayler pour Java, Bamboo Prevalence pour .NET), mais en construire un en utilisant la réflexion est un projet tout à fait réaliste. En .NET 4.0, on dispose d'ailleurs de la Dynamic Language Runtime, qui est typiquement intéressante dans ce cas de figure, pour ne pas trop complexifier le code.

### Limitation sur le volume de données

Une application prévalente utilisant un modèle complètement en mémoire, cela signifie qu'un contexte de prévalence est limité à la mémoire disponible sur une machine. On parle bien entendu de RAM, car dès lors que le swap est utilisé, la perte de performance est énorme : on repasse sur des vitesses typiques des I/O sur disques. A l'inverse, la taille d'une base de données dépendant de celle des disques sur lesquels sont placés ses fichiers, on ne peut pas parler de limite, un ensemble SAN pouvant embarquer plusieurs Po (1024 To).

Evidemment, avec des serveurs de moyenne gamme supportant aujourd'hui jusqu'à 192 Go, vous avez déjà largement de quoi faire. Au pire, vous pouvez séparer des volumes conséquents de données en ensembles relativement autonomes et les faire cohabiter sur des serveurs en parallèle. Et dans le cas particulier où une activité est saisonnière, vous pouvez faire également des roulements sur plusieurs contextes plus petits. La seule limitation est alors le temps de chargement du moteur de prévalence.

Une remarque au passage sur la taille des bases de données : une limitation peut parfois être une bonne chose. Un sondage sur les tailles estimées des Data Warehouses montre que 30% des grands utilisateurs estiment atteindre plus de 10 To dans trois ans (source : lien SlideShare en référence). Honnêtement, qui peut prétendre que ces 10<sup>13</sup> octets sont exploités ? Pour comparaison, le génome humain, qui est une source de données unifiées parmi les plus imposantes, contient 3,4 milliards de paires de base, qui n'ont que quatre combinaisons possibles. Bref, quatre paires par octets, et le tout tient sur 850 Mo. Mais plus on dispose d'espace, plus on l'occupe.

Le problème, particulièrement prégnant pour la Business Intelligence, est que le facteur de compétitivité est la qualité de la donnée, et pas sa quantité. Cette dernière est même un facteur



négatif, par le fait qu'elle ralentit les traitements. Bref, ne rejetez pas la prévalence parce que votre base de données est trop grosse : il y a 99% de chances que les redondances et les niveaux d'agrégation inutiles soient énormes si votre base dépasse 192 Go, et le passage à un modèle prévalent peut vous aider à les supprimer.

### Le facteur psychologique

Enfin, le dernier ennemi de la diffusion de la prévalence est peut-être elle-même, ou en tout cas les sentiments mêlés qu'elle provoque. L'auteur, lors de présentations de cette technologie, a souvent entendu des jugements tels que « *Si la prévalence avait vraiment toutes ces qualités, ça fait longtemps qu'elle remplacerait les bases de données* ». Mais le passé fourmille pourtant de ces technologies maudites. Peut-être entendait-on au début des années 80 « *Si Betamax a autant d'avantages sur VHS, il devrait envahir le marché* » ? Et pourtant, c'est bien VHS qui s'est imposé, malgré des caractéristiques techniques inférieures.

Le facteur psychologique tient également au positionnement des données en mémoire. De la même manière que le réflexe de nombreuses personnes est de penser que leurs données sont plus en sécurité sur leur disque dur non sauvegardé que dans le cloud (ce qui est évidemment faux), le réflexe est de ne pas faire confiance à une application stockant toutes ses données en mémoire, même si les logs de commandes, les snapshots, la qualité des OS serveurs modernes, les alimentations redondantes et les onduleurs rendent virtuellement impossible de perdre des données...

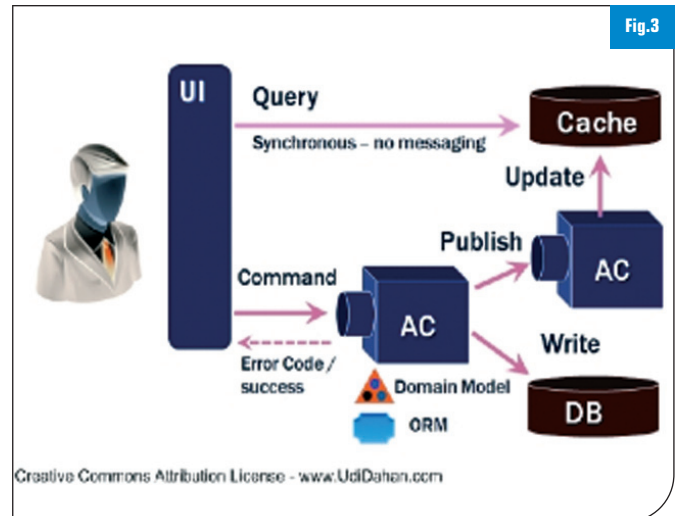
Enfin, comment un moteur de prévalence occupant à peine 47 Ko (taille des deux DLL principales de Bamboo) pourrait rendre des services équivalents à ceux fournis par des moteurs de SGBD pesant 100 Mo en mémoire ? Simplement de la même manière que MySQL a réussi à se frayer une place sur le marché avec des premières versions très pauvres en fonctionnalités (pas de triggers, pas de procédures stockées, etc.), mais très rapides.

Peut-être tout simplement manque-t-il à la prévalence la killer-application qui va déclencher son adoption ? Ou bien peut-être le même type de buzz que pour les bases de données en mémoire, avec SAP pour pousser les feux du marketing ?

## CONCLUSION : PRÉVALENCE ET DDD

Le panorama sur la prévalence ne serait pas complet si on ne parlait pas des liens étroits entre l'approche prévalente et le Domain Driven Design. Le fait de se reposer sur un modèle métier permet de se concentrer sur l'apport d'une compréhension experte du métier à une application, sans perdre du temps avec le mapping O/R, ni toutes les autres technologies « de tuyauterie » sans aucune valeur ajoutée. C'est le principe fondateur du DDD : accorder votre attention et vos meilleurs ressources à votre modèle métier, car c'est votre facteur compétitif. Le reste est simplement de la technique...

En particulier, la prévalence peut être considérée comme une des architectures possibles de mise en œuvre du pattern CQRS (Command and Query Responsibility Segregation). Le principe, en quelques mots, est de séparer les commandes de modification du modèle métier et les requêtes de données en deux bus de traitements bien distincts. Le premier bus se basera bien sûr sur des commandes sérialisées, traitées de manière transactionnelle. Le



second bus pointera sur une ou plusieurs bases de données relationnelles en lecture remplies de manière asynchrone par le modèle de prévalence. Le diagramme ci-dessus (source : lien site web Udi Dahan en référence) montre bien les deux bus.

Dans le cas d'une architecture prévalence, la base de données est bien entendu remplacée par un moteur Bamboo ou Prevayler, et l'O/R M est tout simplement supprimé. Seul compte le modèle du domaine [Fig.3].

Cette décomposition des tâches montre plusieurs avantages :

- On estime que sur des applications de gestion, 20% des besoins de ressources correspondent à des modifications de données et 80% à des lectures. Le fait de dédier les modifications sur un serveur de référence sans lui imposer de requêtes permet une bien meilleure montée en charge.
- L'asynchronisme de mise à jour des données autorise également un lissage des besoins de ressources, avec un impact quasi nul sur la consistance de la donnée. Les lectures fantômes sont rendues impossibles par l'aspect transactionnel des commandes, et le retard de lecture des données fraîches est en général inférieur à un simple décalage humain d'une manipulation.
- La montée en version de l'application de modification des données peut être menée sans impact sur les serveurs de requête, et inversement, à condition d'utiliser des queues de messages comme cela est prévu dans le pattern CQRS.

### Références

<http://blogs.dotnet-france.com/jeanphilippepeg/category/Prevalence.aspx>  
[http://en.wikipedia.org/wiki/System\\_Prevalence](http://en.wikipedia.org/wiki/System_Prevalence)  
<http://bbooprevalence.sourceforge.net/>  
<http://www.slideshare.net/msitpro/sql-server-parallel-data-warehouse>  
<http://www.udidahan.com/2009/12/09/clarified-cqrs/>



■ Jean-Philippe Gouigoux

Architecte logiciel en charge de la R&D chez MGDIS. Il intervient à l'AgileTour et à l'Université de Bretagne Sud pour présenter des méthodes innovantes en architecture et en gestion de l'usine logicielle.

# egilia®

## 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 .



# Atom, AppUp : une nouvelle plateforme pour le développeur !



Intel est connu pour ses processeurs et ses excellents outils de développement multicore. Mais saviez-vous que le fondeur dispose aussi d'une plate-forme complète autour du processeur Atom, allant du netbook à la télévision ? Et qu'il existe une boutique d'applications dédiée aux logiciels Atom ?

Dans ce dossier nous allons vous faire découvrir ce monde encore peu connu.

Désormais disponible en France depuis début décembre 2010, AppUp est avant tout un framework permettant aux développeurs de soumettre des applications pour les différents terminaux. Il accepte les applications Windows et Moblin. Meego est présent aussi. Pour mémoire Meego est une fusion entre Moblin et Maemo et le résultat du travail commun de Nokia et d'Intel.

A vous de jouer !

■ François Tonic



# Tout ce que vous avez toujours voulu savoir sur AppUp

AppUp, la boutique applicative d'Intel est disponible depuis janvier 2010 aux Etats Unis et plus récemment en France. Elle facilite la vente et la diffusion de logiciels C, C++ pour la plateforme Atom d'Intel (netbook, et tout autre matériel Atom). Un peu comme sur le modèle des App Store d'Apple (iOS et plus récemment Mac App Store), Intel veut dynamiser son offre et faciliter la vie des utilisateurs.

Le cœur de AppUp est AppUp Center où l'utilisateur peut voir l'ensemble des applications disponibles et chercher celle qu'il lui faut. Le but est de réduire au maximum le temps passé à télécharger et à déployer. Bref, la distribution des logiciels devient de plus en plus dématérialisée. Le marché de AppUp est très large car il s'appuie sur l'ensemble de l'écosystème Atom : Netbooks, la télé, les tablettes ou autres appareils électroniques intelligents. Et AppUp gère tout cela.

Comme pour Mac App Store, il faut installer l'application AppUp (basée sur Adobe Air) depuis le portail. AppUp est avant tout un framework pour les app store. Il permet aux développeurs de soumettre les applications ou même les composants qu'ils ont créés pour plusieurs appareils utilisant des systèmes d'exploitation différents. D'un autre côté, les consommateurs pourront installer l'AppUp Store sur différents systèmes afin d'accéder au moment souhaité à toutes les applications déjà achetées pour l'utiliser sur n'importe quel appareil, précise Slim Soussi (EMAE SSG Scale program, community manager Intel AppSM developer program, Intel). D'autre part, si AppUp fut au départ très orienté C et C++, Intel a rajouté rapidement d'autres technologies et langages : .Net, Adobe Air et Silverlight et Java sont prévus dans les prochains mois. De quoi réjouir les développeurs. Surtout, pour les applications Windows, AppUp autorise l'implémentation de wrapper. Et comme dit en introduction à ce dossier, AppUp supporte deux systèmes : Windows et Moblin / Meego.



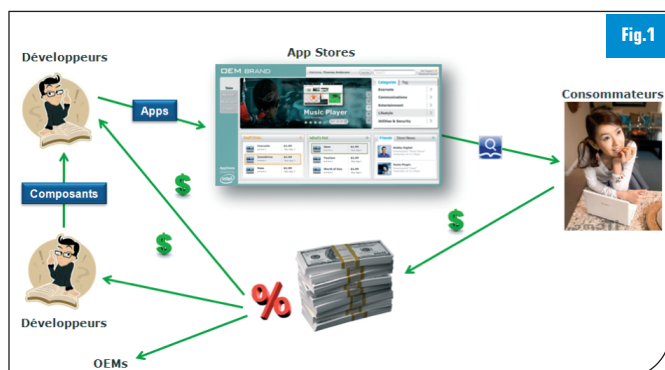
## Business Model et monétisation

AppUp mène plusieurs objectifs : faciliter le développement d'applications pour la plateforme Atom, aider les utilisateurs à trouver la bonne application, aider le développeur à concevoir et à diffuser des applications et enfin, à définir un business model, une monétisation. Sur AppUp, on trouve des applications gratuites et payantes. Le schéma suivant donne une idée de la chaîne de monétisation et du modèle AppUp : [Fig.1].

Bien entendu, l'application doit répondre à certains critères de validation et d'exigences.

Exigences par rapport aux applications et composants applicatifs :

- Votre code doit satisfaire aux critères énoncés dans le document Directives concernant la validation.
- Pour les applications, nous vous recommandons d'utiliser les composants publiés dans le catalogue développeur lorsqu'ils conviennent, pour accélérer le développement et réduire les risques de rejet lors du processus de validation.
- L'utilisation du kit SDK du programme destiné aux développeurs Intel AppUpSM est exigée, sauf pour les applications code source libre.
- Utilisez le GUID conformément aux instructions de l'API fournies dans le kit SDK.
- Testez soigneusement votre code avant de le soumettre. Si vous développez pour le système d'exploitation Moblin\*, utilisez les outils de test de conformité Moblin, accessibles à Moblin.org.
- Emballez votre application conformément au Guide d'emballage des applications.
- Lorsque vous soumettez une application, suivez les instructions de soumission et fournissez des informations appropriées et



précises dans les formulaires de soumission.

- Nous vous recommandons d'utiliser des numéros de version.

## Quelques exigences de validation :

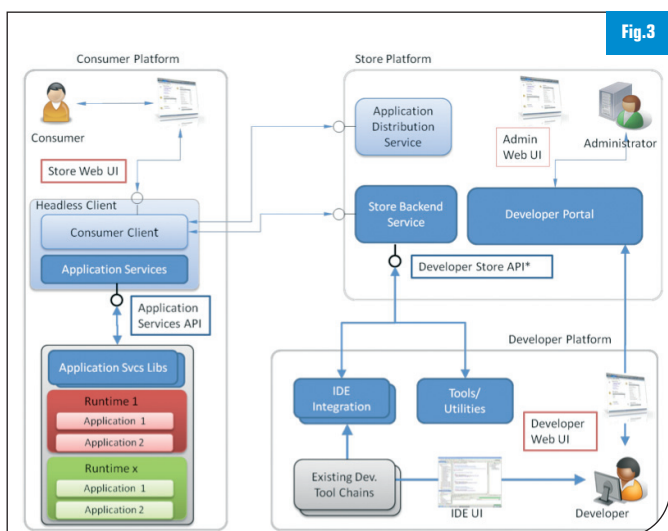
- Veuillez à posséder les autorisations vous permettant d'utiliser les marques de commerce d'autres sociétés.
- Veuillez à posséder les droits vous permettant de distribuer votre application. Seules les applications code source libre doivent soumettre le code source.
- Veuillez à ce que votre application s'installe et s'exécute correctement ; et qu'elle se désinstalle proprement.
- Veuillez à ce que votre interface utilisateur soit propre et cohérente.
- Veuillez à classifier votre application correctement lors de sa soumission et incluez tous les critères requis dans les formulaires de soumission.
- N'autorisez aucun comportement pouvant nuire à d'autres logiciels ou réseaux.
- N'autorisez aucun comportement collectant et utilisant des informations personnelles sans autorisation explicite de l'utilisateur.
- N'incluez aucun contenu répréhensible.
- N'incluez pas de fenêtres publicitaires ni de publicités cherchant à vendre ou à mettre à niveau en dehors de la boutique.

Vous devrez respecter une liste précise de recommandations, consultez le guideline officiel.

<http://appdeveloper.intel.com/fr-fr/article/validation-guidelines>

**Fig.2**

	Anonyme Gratuit	Avec un ID Gratuit <a href="#">Obtenir ID</a>	Membre du programme 99-\$ par an Gratuit <a href="#">S'inscrire</a>
Télécharger des outils et des ressources pour développeurs	✓	✓	✓
Participer aux blogs	✓	✓	✓
Parcourir les forums	✓	✓	✓
Participer aux forums		✓	✓
Télécharger des composants		✓	✓
Examiner et noter des composants		✓	✓
Télécharger le SDK		✓	✓
Gagner des points de réputation		✓	✓
Obtenir un support		✓	✓
Soumettre des composants pour le catalogue développeurs			✓
Soumettre des applications pour l'App Store			✓



Trois modes de diffusion sont disponibles : anonyme (sans ID), avec un ID et gratuit, membre du programme (actuellement gratuit, au lieu, de 99 \$). Voici le résumé : [Fig.2](#).

## Le modèle de développement

Redonnons les fondamentaux d'AppUp. Tout d'abord, AppUp se destine à toutes les plateformes Atom et fonctionne sur Windows, Moblin et son successeur, Meego. Très clairement, Moblin n'est plus un système mobile de référence pour Intel. Des SDK sont disponibles pour l'ensemble de ces systèmes (C et C++). La technologie Air d'Adobe est elle aussi disponible et possède son propre SDK. A noter que le SDK Meego n'est pas en version finale [Fig.3](#). En C++, Intel définit deux méthodes de base pour créer son application : en ajoutant le framework AppUp, définir une nouvelle application par une sous-classe (ou subclassing) à la classe application. Cette deuxième approche est celle préconisée par Intel. Si vous passez les API C, vous initialisez la librairie (et n'oubliez pas de préciser votre ID). Le code élémentaire ressemble à ceci :

```
#include <stdio.h>
#include "adpcore.h"
int main( int argc, char* argv[] )
{
    ADP_RET_CODE ret_code;
    const ADP_APPLICATIONID myApplicationID = {{
        0x12345678, 0x11112222, 0x33331234, 0x567890ab}};
```

D'autre part, les SDK et API évoluent régulièrement, des mécanismes de versions sont disponibles. En C++ cela donnera :

```
namespace com::intel::adp
const wchar_t* ADP_API_VERSION;
const unsigned long ADP_API_LEVEL;
```

Avec AppUp, les applications s'exécutent de deux manières : nativement (C, C++) et via un runtime (Air). Mais Intel travaille à supporter de nouvelles technologies : Silverlight et Java !

N'oubliez pas que votre application AppUp est susceptible de fonctionner sur des terminaux divers et variés donc avec des contraintes matérielles différentes, notamment l'écran. Il faut absolument adopter l'indépendance des couches. Sur les problématiques matérielles, reportez-vous à l'article développement sur la plateforme Atom.

## L'indispensable ID

Avant toute chose, le développeur doit s'inscrire au programme Intel AppUp. Il donne accès au portail développeur, aux SDK, documentations, exemples de code, etc. Cela vous permettra d'obtenir un numéro identifiant ou ID. Cet ID est important car il vous permettra de charger et d'utiliser les composants logiciels pour vos développements. Dans le modèle de développement AppUp, vous pouvez utiliser des composants d'application disponibles dans le catalogue développeur.

A noter qu'il existe plusieurs types d'ID. Pour les ID des applications et des composants, il s'agit d'une clé de 128-bit. Et l'identification par la plateforme AppUp de son logiciel passe obligatoirement par un ID. D'autre part, il existe un ID de production (pour soumettre votre application et la déposer sur le store)

et un ID de debug. Un ID Debug autorise uniquement une exécution pour tests via la *Application Test and Design Service*. En aucun cas, un ID Debug n'autorise une soumission au store.

## Adobe AIR : c'est facile !

Une des forces d'AppUp est son ouverture technique sur plusieurs langages et runtime d'exécution. L'arrivée de AIR sur la plateforme constitue un avantage non négligeable pour les développeurs. Car finalement, au lieu d'apprendre C ou C++, si vous ne les connaissez pas, vous pouvez rapidement utiliser ActionScript / MXML et la plateforme de développement Adobe Air. Pour ce faire, un SDK spécifique est disponible : Melrose.

Pour permettre l'intégration des applications AIR dans AppUp, on dispose des fonctions propres à la boutique. Ainsi, les packages applicatifs ont été rajoutés. Il faut intégrer le fichier .swc à son projet Air (qu'il soit nouveau ou déjà existant).

Portail officiel : <http://appdeveloper.intel.com/adobe-air>

Guide pour le package Air sur AppUp : <http://appdeveloper.intel.com/en-us/article/adobe-air-packaging-guide-atom-developer-program-submissions>

## Et les codes existants ?

Tout va dépendre du langage et du type de logiciels. Il faudra, si vous souhaitez intégrer AppUp, inclure le SDK et les API propres à

### Composant, vous avez dit composant ?

Le modèle de développement AppUp repose beaucoup sur la notion de composants. En effet, le développeur peut récupérer des composants pour accélérer son propre développement. Inutile donc de se lancer tête baissée sur son projet. Il n'est pas inutile de voir ce que le catalogue composant (component catalog) propose et il y a des choix : wrapper DLL, librairie pour Java, géolocalisation, API et modules de monétisation, module RSS, etc. Ces composants peuvent être en différents langages : C#, MFC, Java.

Surtout, tout développeur AppUp peut déposer son composant. Les pré-requis avant tout dépôt sont :

- Avoir reçu un ID et adhéré au Programme pour développeurs Intel Atom (voir Comment adhérer au Programme pour développeurs Intel Atom).
- Avoir fourni les informations de paiement de votre organisation si vous souhaitez être rétribué pour vos composants.
- Avoir créé un GUID pour votre composant ; vous pouvez créer un GUID lors de la proposition, il doit être utilisé dans votre code.

Avoir créé un fichier binaire compilé pour la proposition (voir les Consignes de conformité et de validation des applications/composants et les Consignes pour les développeurs pour en savoir plus). Le fichier binaire doit inclure la documentation de l'utilisateur de l'API et les informations de licence. Nous vous recommandons d'examiner les exemples de composants et la documentation du SDK du Programme pour développeurs Intel Atom (IADP).

Puis le composant suit son parcours de validateur que le développeur peut visualiser à tout moment depuis un tableau de bord dédié.

Pour en savoir plus : <http://appdeveloper.intel.com/fr-fr/article/how-submit-component>

la plateforme. Si vous prenez une application native iPhone écrite en Objective-C, là pas beaucoup de solutions : réécrire en C++ ou éventuellement C# mais peu de choses à reprendre. Si vous souhaitez avoir une approche non exclusive, Slim Soussi explique ainsi une possible démarche : « *Si vous avez la possibilité d'envisager des efficacités inter plateformes, vous voudrez sans doute créer des bibliothèques communes en C++ susceptibles d'être utilisées par l'application iPhone et par Windows. Le contenu commun pourra être créé en HTML, ce qui lui permettra de s'afficher de manière semblable dans tous les environnements.* ».

L'autre problème peut venir de l'interface et de l'ergonomie de l'application comme nous l'avons brièvement indiquée plus haut. Slim Soussi explique : « *Lorsque vous portez une application d'iPhone vers un netbook, prenez en considération la différence d'expérience utilisateur permise par chacune des plateformes. Tout comme le portage vers l'iPad, les netbooks vont avoir des écrans plus grands et vous devrez mettre votre application à l'échelle. La taille de cet écran autorise une interface plus riche, ce qui permet de combiner en une seule et même vue des tâches en série comme une liste, suivie d'une vue détaillée. Si vous avez déjà une application iPad, cette application peut être transférée directement sur une plateforme netbook Windows. Autre point à examiner : la manière dont votre application iPhone utilise le tactile afin de l'adapter à une interface souris, la plupart des netbooks étant dépourvus d'écran tactile. La principale différence sera le manque de « gestures » (comme le pinch zoom) sur la plupart des netbooks Windows et le fait que vous pouvez ajouter des effets hover lorsque le pointeur de la souris effleure un élément, sans cliquer dessus.* »

Le plus dur sera de bien vérifier les frameworks et API, et ceci est valable pour les autres plateformes que l'iPhone. Par exemple, pour les jeux :

	iPhone	Windows
IU	CoreGraphics	DirectDraw
Langage	Objective-C	C++
Framework	CocoaTouch	MFC

Pour aller plus loin : <http://appdeveloper.intel.com/fr-fr/article/iphone-netbook-porting-technical-guide>

## Bref, une application AppUp c'est...

Une application AppUp ressemble à toute application mais elle nécessite la prise en compte de contraintes propres aux marchés AppUp et donc des terminaux basés sur Atom allant du smartphone aux boîtiers de télévision. A vous déjà de cibler les boutiques AppUp visées et les terminaux Atom.

A partir de là, vous pourrez définir les contraintes de l'interface et de l'ergonomie (si vous êtes en contexte tactile ou non par exemple). Autre contrainte, la taille de l'écran change d'un téléphone à un netbook par exemple, donc, définir une taille d'interface (et donc son agencement) selon chaque terminal. Et vous devrez aussi réfléchir à l'utilisation ou non des fonctions réseaux, de téléphonie, de stockage, etc.

■ François Tonic



## Témoignage

# Un spécialiste du livre numérique se met à AppUp

Readbooks / Mobilire développe une librairie numérique (livres, romans, BD), en se spécialisant sur le ebook, accessible sur PC, smartphone, tablette, et ce, en plusieurs langues. Les tablettes tactiles constituent un marché très important du fait de la multiplication des annonces et de l'offre comme ce fut le cas durant le salon de l'information grand public, CES, à Las Vegas. Yannick Lacoste (PDG Mobilire) nous a rappelé le lien particulier avec Intel : « Intel est intéressé par intégrer à son écosystème le livre numérique, le ebook. Nous travaillons avec eux depuis plus d'un an ».

Intel n'est plus uniquement un fondeur fournissant un processeur mais souhaite réellement s'investir sur l'ensemble des solutions, de la plateforme avec les supports, les App Store, les logiciels. C'est la stratégie, la vision « Intel Continuum ». Intel travaille aux Etats Unis avec des partenaires de contenus. Et ce contenu américain intéresse tout naturellement notre éditeur français pour enrichir sa propre offre. « AppUp n'est pas très connu » s'amuse à rappeler Y. Lacoste.

Cependant, comme Intel a une vision à moyen et long terme, ce n'est pas un réel handicap, c'est un travail de longue haleine avec un marché en pleine explosion à court terme. L'un des objectifs du fondeur est de proposer AppUp au marché OEM. « AppUp est une marque blanche. C'est-à-dire que des constructeurs peuvent l'intégrer, comme l'Asus Store (annoncé en juin 2010, NDLR) », précise Y. Lacoste. C'est là un avantage non négligeable : avoir des Intel AppUp directement pré-installés sur les machines, ce qui n'est pas le cas aujourd'hui, car il faut télécharger et installer un logiciel client AppUp.

## Du choix et des choix

Avec la multiplication des boutiques applicatives, il faut faire le tri entre ce qui fonctionne, ce qui ne fonctionne pas, et les boutiques prometteuses. Et ensuite il faut faire des choix de plateformes, de terminaux. Ainsi si AppUp propose des SDK pour Windows, Moblin, Meego, Mobilire a une approche pragmatique : « Moblin c'est le passé ! L'avenir est (clairement) Meego même s'il s'agit, encore d'un challenge. Mais nous misons beaucoup sur lui ! ».

Et si Meego reste en développement et peu présent, il s'agit d'un système important pour l'éditeur qui travaille dessus depuis quelques mois.

Mais au-delà du choix matériel et système, un autre écueil peut intervenir pour les éditeurs. Et Readbooks / Mobilire y a été très sensible. « Le fait que AppUp supporte et soit compatible avec Adobe Air a été quelque chose de fondamental pour nous. Nous n'aurions pas pu développer nos logiciels et contenus aussi rapidement.

Et Intel n'est pas le seul à miser sur Air », s'enthousiasme Y. Lacoste. Le développeur bénéficie du moteur WebKit, d'ActionScript.

« Pour nous, c'est facile de développer avec Air. Nous disposons aussi d'un SDK dédié, Melrose Project. Ceci dit, nous disposons d'un langage d'assez haut niveau ».

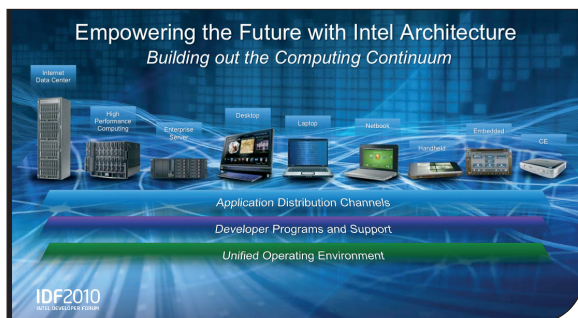
Et surtout, les applications développées en Air fonctionneront sur d'autres systèmes que AppUp. Ainsi RIM



a dévoilé un SDK Adobe Air pour son Playbook. Une des stratégies de Readbooks / Mobilire est de pouvoir être pré-installé et de pouvoir proposer directement aux utilisateurs la librairie Mobilire sur leur machine. C'est un modèle souple et bien moins contraignant qu'un Amazon par exemple qui sera plus rigide sur les conditions d'accès. « Nous mettons en place des partenariats gagnant-gagnant avec les constructeurs », explique Y. Lacoste.

Et pour Readbooks / Mobilire, les App Store font partie des habitudes des utilisateurs. « Nous y croyons beaucoup. Et nous voyons les effets quand une (nouvelle) application est sur un App Store. Et plus ça va, plus ces boutiques se multiplient, s'étendent », conclut M. Lacoste. Et d'ailleurs AppUp n'est pas le seul à être décliné. Ainsi le market Android se retrouve sous une nouvelle forme chez Archos.

■ François Tonic



# 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.*



Actus / Événements / Newsletter / Vidéos

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

☐ **OUI, je m'abonne** (écrire en lettres capitales)

Envoyer par la poste à : Solutions Logiciels, service Diffusion, 22 rue René Boulanger, 75472 PARIS - ou par fax : 01 55 56 70 20

**1 an : 50€ au lieu de 60€, prix au numéro** (Tarif France métropolitaine) - Autres destinations : CEE et Suisse : 60€ - Algérie, Maroc, Tunisie : 60€ , Canada : 80€ - Dom : 75€ Tom : 100€  
10 numéros par an.

☐ M. ☐ Mme ☐ Mlle Société .....

Titre : ..... Fonction : ☐ Directeur informatique ☐ Responsable informatique ☐ Chef de projet ☐ Admin ☐ Autre .....

NOM ..... Prénom .....

N° ..... rue .....

Complément .....

Code postal : [ ] [ ] [ ] [ ] Ville .....

Adresse mail .....

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

# Développement de logiciels pour la plateforme Intel Atom



La plateforme Intel Atom inclut le matériel et le logiciel adaptés aux matériels compacts. Ceux-ci vont des netbooks, aux tablettes, en passant par les systèmes embarqués tels que les systèmes d'« Infotainment » pour véhicules et les SmartTV. Le matériel est une combinaison des dernières avancées en termes de micro-processeurs Intel, de chipsets et de composants matériels, tels que le Wifi, la 3G, l'accélération vidéo et graphiques 3D, tous conçus pour assurer une consommation électrique aussi réduite que possible.

**L**a plupart des plateformes Atom, y compris les netbooks à processeur Intel, supportent pleinement le système d'exploitation Windows. Cela signifie que toutes vos applications préférées, avec quelques ajustements pour la taille de l'écran, devraient être disponibles sur les plateformes Intel Atom. Sur certains segments de marché, la plateforme inclut le système d'exploitation MeeGo. Intel, en partenariat avec Nokia, a investi des ressources importantes dans ce système open source. MeeGo, hébergé par la Linux Foundation, comprend un noyau Linux optimisé pour la plateforme ainsi que du middleware et des logiciels d'application.

## Pourquoi développer pour Atom?

La plate-forme a un certain nombre d'avantages importants aussi bien techniques que commerciaux. Comme toutes les plateformes sont construites sur la même architecture, les applications peuvent être adaptées plus rapidement à l'ensemble des terminaux Atom. Cette (micro)architecture est identique à celle des ordinateurs Intel x86. Ce rapprochement permet à l'ensemble de l'écosystème de tirer profit de la vaste gamme logicielle existante. Bien que la plateforme soit conçue pour une autonomie maximale, un haut niveau de performance a été maintenu. Par exemple, les applications bénéficient du décodage et de l'encodage de vidéo HD

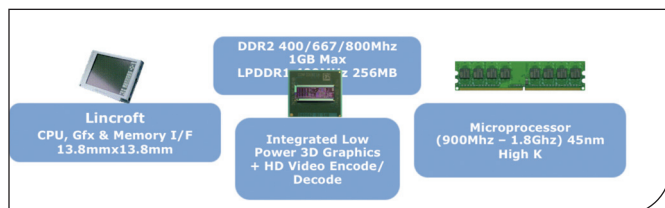


par le matériel, du support d'exécution en mode multitâche, grâce la technologie Intel Hyper-Threading, ainsi que d'un niveau élevé de performance avec la technologie Intel Burst Performance. Depuis toujours, les appareils mobiles ont eu du mal à satisfaire les attentes des utilisateurs pour l'accès aux contenus internet. Dès le début, la plateforme Atom amène un support optimal de ces technologies. Aussi disponible sur la plupart des plateformes Atom, Intel AppUp. Il s'agit d'une boutique en ligne d'applications (voir les autres articles du dossier, NDLR). La publication de votre application dans cette boutique vous permettra de profiter d'une base installée en croissance constante et de l'extension de la boutique en ligne sur de nouveaux segments de marché.

## Caractéristiques des technologies et des composants

L'élément central est le processeur Atom. Dans ce design de type System On Chip (SoC), le packaging du processeur contient les composants pour le support, au niveau matériel, du décodage 1080p et l'encodage 720p (H.264, MPEG4 notamment), du 2D/3D (OpenGL ES 2.0 et OpenVG 1.0) pour le graphisme, et un contrôleur de mémoire (DDR2 à 800 <MT/s> and LPDDR1 à 400 <MT/s>). Les composants de communication sont des modules pour le Wifi, le Bluetooth, le WWAN et le WiMAX. Les cas d'usage liés à la mobilité sont supportés grâce à la présence de caméras à haute résolution, de SDIO, de USB OTG et de la gestion optimisée de la consommation électrique liée à l'audio. D'importants progrès d'économie d'énergie ont été réalisés sur le processeur, le chipset et les autres composants, y compris la technologie Enhanced Intel SpeedStep.

Cette technologie permet aux systèmes d'exploitation de piloter le processeur pour l'amener vers des fréquences et des tensions plus basses lors de l'exécution d'une charge de travail donnée. En plus, de nouveaux « états de sommeil » du processeur, plus profonds que C6, ainsi que des fonctions de « power gating » sur les îlots électriques du circuit, amènent une réduction significative de la puissance consommée dans les états d'inactivité par rapport à la génération précédente. D'autres composants comprennent le support de fonctions de sécurité via l'accélération matérielle d'algorithmes cryptographiques, le support d'I/O standard, de composants USB, de contrôleurs de systèmes intégrés, de contrôleurs de NAND et de générateurs d'horloge. Une réduction significative de la taille du packaging par rapport aux précédentes générations a été rendue possible grâce au processus de fabrication en 45nm et la technologie Intel Hafnium basé sur les transistors « High-k + metal gate ».



## TECHNIQUES D'OPTIMISATION

### Optimisations de l'interface utilisateur

Atom ne nécessite pas, en tant que telle, une interface utilisateur particulière ou un paradigme de navigation spécifique. Cependant, il y a un certain nombre de principes de design et d'interface /

ergonomie qu'il est bon de garder à l'esprit. L'existence d'une vaste gamme d'appareils signifie que votre application sera confrontée à différents environnements en termes de taille d'écran, de résolution, d'orientation d'écran, et de forme. Plus votre interface utilisateur sera flexible au niveau de la taille d'écran et de la mise en page, plus vite l'application pourra être disponible. Les éléments de l'interface peuvent être redimensionnés et intelligemment remaniés pour faire un meilleur usage de l'espace disponible. Il est important de rendre l'interface utilisateur simple et facile à utiliser. Soyez bref et concis. En outre, le principe de conception qui consiste à séparer la présentation, de la logique et des données est importante. Sur les appareils disposant d'un écran tactile, pensez à l'ergonomie et à la taille des doigts pour le design des éléments de l'interface. L'élément le plus important est de rendre évidente la fonction de votre application. Réfléchissez à ce qui pourrait inciter à utiliser votre application.

### Gestion de l'énergie

Les plateformes mobiles ont besoin de logiciels prenant en compte la consommation d'énergie. La meilleure chose qu'un logiciel puisse faire est de permettre au processeur de « dormir » aussi souvent et aussi longtemps que possible. L'optimisation d'un algorithme ou d'une charge de travail, en réduisant le nombre de cycles requis pour effectuer ce travail, permet de mettre le processeur dans un état de basse consommation plus tôt. Éviter les opérations ou les programmes qui « réveillent » souvent le matériel, comme les boucles d'attente ou le « polling ».

Les applications peuvent également prolonger la vie de la batterie en réduisant leur besoin de performances ou leur niveau de fonctionnalités. Cela peut sembler étrange au premier abord, mais en réduisant certaines sections de programmes effectuant des calculs intensifs, sans compromettre toutefois sa fonctionnalité première, cela peut avoir un effet positif sur la durée de vie de la batterie et de fait, sur l'expérience utilisateur. Voici des exemples de ce qui pourrait être fait : fermer les activités d'arrière-plan, diminuer la qualité, la profondeur des couleurs, ou la résolution de la vidéo et de l'audio, réduire la fréquence des activités de journalisation, réduire la quantité ou la qualité des animations, limiter l'analyse ou l'indexation de fichiers. De même, il est conseillé de différer toute activité nécessitant de la puissance de calcul jusqu'à ce que l'appareil soit branché sur secteur.

### Optimisation

Une des façons les plus rapides et les plus simples d'obtenir un gain de performance sur les plateformes Atom est d'utiliser le compilateur de C++ d'Intel. Il génère du code incluant de multiples optimisations spécifiques aux processeurs Atom. Cela inclut des optimisations telles que la prise en compte du mode d'exécution « in-order », de manière à améliorer l'autonomie du système. Le compilateur analyse le code et réorganise les instructions de façon à supprimer les latences. Pour profiter de ces optimisations, ajoutez le compilateur à votre environnement, puis simplement utilisez l'option du compilateur '-xSSE3\_ATOM' or '-QxSSE3\_Atom'.

### SSE3

Pour des traitements logiciels très lourds en termes de données, une amélioration significative des performances peut être obtenue

en utilisant les extensions Streaming SIMD Extensions 3 (SSE3). Il s'agit d'un ensemble d'instructions standard, compatibles avec tous les processeurs x86 récents. En outre, Atom prend également en charge les instructions SSSE3 (le premier S pour supplémentaire). SSSE3 est particulièrement intéressant pour les développeurs de jeux en raison de l'ajout d'une opération de produit scalaire, qui fournit un meilleur rendement lors des opérations à virgule flottante.

```
_asm {
    //— Setup —
    push ebx
    mov eax,pC0 // xmm7 - load the two Cs: Cli Clr C0i C0r
    movups xmm7,[eax]
    // xmm6 - four copies of exit test: 4.0 4.0 4.0 4.0
    mov ecx,pFours
    movups xmm6,[ecx]
    // xmm4 - counter - using negatives and initialized to one
    as 'C' asm count each time thru
    //pcmpeqw xmm4,xmm4 // FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
    pcmpeqd xmm4,xmm4 // FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
    // xmm1 - initial Z values Z1 and Z0: d c b a or i1 r1 i0 r0
    pxor xmm1,xmm1
    xor ecx,ecx // loop counter

    //— Test Pixels —
TestPixels:
    add ecx,1 // max iteration counter
    // partial results
    movsldup xmm0,xmm1 // c c a a - real pairs
    movshdup xmm2,xmm1 // d d b b - imaginary pairs
    mulps xmm0,xmm1 // dc cc ba aa - ri ii ri rr
    shufps xmm1,xmm1,0xB1 // c d a b - reordered 'Z' value
    mulps xmm2,xmm1 // dc dd ba bb - ri rr ri rr
    // finish the complex multiply
    addsubps xmm0,xmm2 // dc+cd cc-dd ba+ba aa-bb - d' c' b' a'
    // add constant C with result back in xmm1
    movaps xmm1,xmm7
    addps xmm1,xmm0 // Z = Z + C
    // length sq
    movaps xmm3,xmm1 // copy of 'Z'
    mulps xmm3,xmm1 // dd cc bb aa
    haddps xmm3,xmm3 // dd+cc bb+aa dd+cc bb+aa

    //— Loop Control —
    cmpltps xmm3,xmm6 // L1<4.0 L0<4.0 L1<4.0 L0<4.0
    paddb xmm4,xmm3 // add the counters
    movmskps eax,xmm3 // extract sign bits
    cmp eax,0
    je Done
    cmp ecx,255
    jl TestPixels

    //— Counters —
Done:
    movd edx,xmm4 // Count0
    pshufd xmm4,xmm4,0xB1 // reordered
    neg edx // we counted negatives
}
```

```
mov eax,pCount0
movd ebx,xmm4 // Count1
mov [eax],edx
neg ebx
mov ecx,pCount1
mov [ecx],ebx

pop ebx //— Cleanup —
}
```

## Intel Hyper-Threading Technology (HTT)

HTT, qui permet à un seul processeur d'apparaître et d'agir comme deux processeurs, est une caractéristique importante de la plateforme. Il offre un gain de performances particulièrement utile sur ce processeur qui dispose d'un mode d'exécution « in-order », en évitant les goulets d'étranglement. Les avantages du multitâche sont évidents car il permet au système de basculer d'une tâche à une autre de manière plus réactive. En raison des atouts apportés par le multitâche sur la plateforme, il est souhaitable et avantageux pour les logiciels de mettre en œuvre une architecture multithreading.

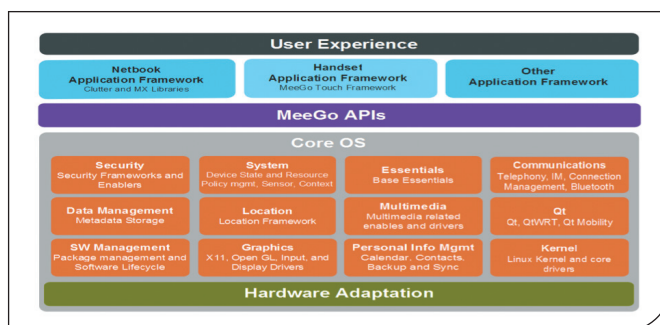
## La suite des outils Intel

La suite d'outils de développement de logiciels pour Atom, cible l'analyse des performances du logiciel sur les plateformes embarquée, les netbooks, et les appareils électroniques grand public. La suite d'outils couvre tout le cycle de développement des logiciels. Elle est disponible sur Windows et Linux, et comprend : le compilateur C++ de Intel pour Linux, le débogueur d'application pour le processeur Intel Atom, des bibliothèques de fonctions optimisées (IPP) pour Linux et l'analyseur de performances VTune pour Linux.

## OPTIMISATION DES JEUX SUR LES CHIPSETS DE NETBOOK

Orion Granatir et Omar Rodriguez, ingénieurs logiciels au sein du Visual Computing Group d'Intel, abordent les problématiques d'optimisation des jeux sur les chipsets des netbooks. Trois accélérateurs graphiques / média sont disponibles : GMA 945, 3150 et 500. Ils jouent un rôle important dans l'optimisation des performances pour les jeux sur plateforme Atom. Sur les séries N400 de processeurs Atom, on dispose d'une intégration du contrôleur mémoire et du contrôleur graphique.

Les trois chipsets utilisent l'accélération matérielle de DirectX 9.0c. **Rappel** : si vous utilisez les API DirectX, pour le moment, vous êtes limité à la version 9.0. D'autre part, une distinction existe : pour les traitements des vertex sur le processeur depuis un chipset GMA 945 ou 3150, il n'y aura pas d'accélération maté-



rielle sur ces traitements. Par contre, le GMA 500 aura un tel support. Ce chipset est aussi compatible avec DirectX 9.0c et fournira de meilleures performances. Une bonne stratégie est de valider et de tester sur les GMA 945 et 3150, tout en faisant une validation, du moins des tests, sur une machine basée sur un GMA 500.

Autre point important à considérer : le balancement des traitements sur le CPU et le GPU. Pour vous aider, utilisez le Intel Graphics Performance Analyzers (ou Intel GPA). Lorsque vous exécutez une lourde charge de travail graphique, utilisez la technologie HTT. Et pour supporter plusieurs threads et en rajouter, vous pourrez utiliser l'outil Intel Threading Building Blocks.

Cette approche, de threads multiples, est avantageuse si vous chargez des assets et de nouvelles scènes. Vous pourrez ainsi vérifier la charge des threads et les performances, car le threading ampute toujours ces dernières. Si vous lancez plusieurs threads, les performances graphiques se dégraderont. Il faudra alors du temps pour changer de contexte (d'un thread à un autre).

Quand vous utilisez un GMA 945, ou 3150, pour optimiser tout cela, vous pouvez appliquer les techniques suivantes :

- compression de la texture
- minimiser le post-traitement
- réduire le niveau de données envoyées aux processeurs

D'autres astuces sont possibles : utiliser des tampons, "triangle strips" pour réduire les vertex traités par le chipset.

Quand vous développez pour des netbooks, diverses bonnes pratiques vous aideront à composer avec des écrans de petite taille : garder à l'écran uniquement les éléments d'interface, de jouabilité, nécessaires pour jouer à l'instant T. Utilisez des icônes reconnaissables car vous devrez supprimer le texte ou alors utiliser le "tool-tip" du bouton, de l'objet d'interface.

Il est très important pour votre jeu de considérer la durée de vie de la batterie. Le netbook a en général une autonomie supérieure à celle des portables classiques. Cependant, les jeux ont tendance à réduire l'autonomie plus rapidement que les autres applications (notamment à cause des exigences graphiques, ndr). Pour aider le développeur, Intel a conçu le Laptop Gaming Technology Development Kit (TDK) qui permettra de monitorer, de surveiller le niveau des batteries, le réseau, et l'état de la plateforme. Si vous avez besoin d'un contrôle plus fin de l'énergie, vous pouvez utiliser les fonctions du gestionnaire d'énergie de Windows (ex. : hibernation, suspension d'activité...).

TDK fournit également des fonctions pour créer une zone de réseau personnel (autour du netbook). Cette fonction évite de créer un réseau entier. Le Peer Windows Collaboration API fournit des fonctionnalités pour trouver d'autres personnes sur le réseau, même si elle oblige les utilisateurs à être connecté au même sous-réseau ou réseau local.

## Le système MeeGo

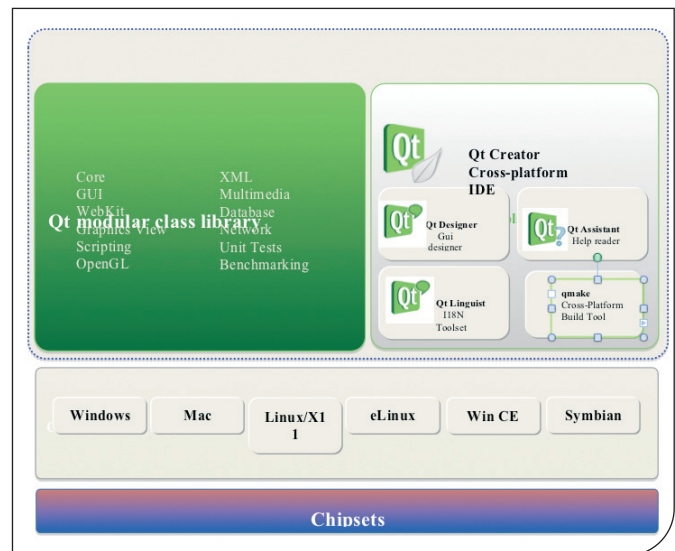
MeeGo est une plate-forme Linux standard optimisée pour les plateformes Intel Atom et les usages mobiles. Avec le noyau Linux, il comprend d'autres composants « open source » fournissant des fonctionnalités de communications, de multimédia, de graphiques 3D et d'autres caractéristiques importantes pour les appareils mobiles. Au-dessus des composants « middleware », il y a une couche directement liée à l'expérience utilisateur, composée de

l'interface utilisateur et des applications de base. Cette couche est adaptée selon les besoins des différents segments de marché. Voici un schéma de l'architecture MeeGo :

## QT: un "framework" applicatif

MeeGo inclut la plateforme de développement et d'exécution d'applications QT ainsi que son SDK. Elle supporte les versions les plus récentes de Qt Quick et Qt Mobility. Grâce à l'intégration de Qt, vous pouvez créer des applications QT ciblant les terminaux basés sur MeeGo.

qmake  
Cross-Platform  
Build Tool



## MeeGo SDK

Le MeeGo SDK fournit un environnement basé sur Qt Creator pour le développement, le débogage et l'exécution d'applications mobiles pour les plateformes Atom. Un émulateur (QEMU) est également disponible. Il permet de tester les applications en simulant le système cible.

## LE DÉPLOIEMENT D'APPLICATIONS AVEC INTEL APPUP

AppUp est à la fois une boutique d'applications et un programme de développement pour les appareils à base d'Intel Atom. AppUp permet aux développeurs de distribuer et vendre des applications aux utilisateurs de systèmes basés sur Atom. Le programme développeurs AppUp fournit toutes les ressources nécessaires pour préparer les applications permettant le déploiement dans la boutique AppUp.

Pour les éditeurs de logiciels, AppUp offre l'opportunité de travailler en partenariat avec Intel et fournit un canal de distribution additionnel ainsi qu'une source de revenus supplémentaires. Pour les développeurs, AppUp fournit un canal de distribution facilement accessible. Le modèle économique permet aux développeurs de fixer un prix pour leurs applications et de recevoir jusqu'à 70% des recettes de chaque vente. À l'heure actuelle, AppUp est compatible avec Microsoft Windows, Moblin et dispose d'une version bêta 2 pour MeeGo.



## Inscrivez-vous au programme de développeur Intel AppUp

Le AppUp SDK contient l'API de référence, un guide développeur, des exemples et un débogueur. Il est actuellement disponible pour les langages suivants: C, C++, .NET, Java et Adobe AIR. L'architecture d'Intel AppUp se compose d'un agent client qui s'exécute sur le système de l'utilisateur et d'une plateforme de service AppUp Center. Ces deux composantes coopèrent pour permettre à l'utilisateur de naviguer sur le magasin, d'acheter des applications, de les installer et d'autoriser l'utilisation de ces applications. Le schéma suivant illustre les interactions entre l'utilisateur, le développeur et l'administrateur AppUp ... L'Intel AppUpSDK fournit une API simple qui permet au démarrage de l'application de communiquer avec la plateforme Intel AppUp Center, de manière à autoriser l'exécution de cette application. Elle permet aussi le cas échéant, de gérer le traitement des erreurs ou d'introduire d'autres instrumentations optionnelles. L'extrait de code suivant illustre une section de code typique :

```
// An AppUp application ID is obtained from Intel and unique to
the product const ADP_APPLICATIONID myAppID = {0x11111111,
0x11111111, 0x11111111, 0x11111111};

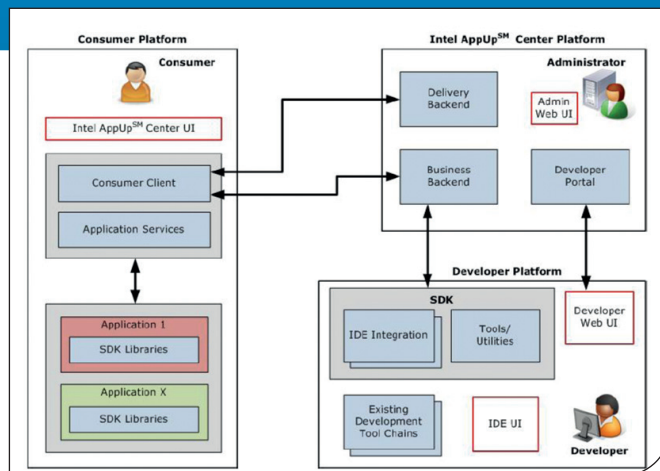
// Initialize AppUp
ADP_Initialize();

// Check the application is authorized to run.
if (ADP_IsAuthorized(myAppID == ADP_AUTHORIZED) {
    //your application code goes here

// Cleanup adpcore data structure, =shutdown connections used
by the adpcore.
ADP_Close();
}
```

## Soumettre l'application et compléter le processus de validation

Après que l'Intel AppUp SDK ait été intégré dans une application et testé avec succès, l'application est soumise au processus de validation. Celui-ci consiste à vérifier l'intégration de l'Intel AppUp SDK de façon à garantir une installation fiable et un fonctionnement correct de l'application. Il permet aussi de s'assurer de l'adéquation des informations correspondantes dans le catalogue.



## Conclusion

Avec la troisième génération de produits basés sur plateforme Atom, les matériels, logiciels, outils de développement et canaux de distributions forment un environnement complet pour aider les développeurs de logiciels à créer des applications compétitives pour les appareils mobiles. Que vos produits soient des applications multimédia, des outils de communications, des jeux ou des logiciels professionnels, Intel Atom est la plateforme idéale pour vous associer au développement du monde de l'informatique mobile connectée.

## Pour plus d'informations

- <http://software.intel.com/fr-fr/>
- Guide de développement pour Intel Atom (en anglais) : <http://software.intel.com/en-us/articles/developers-guide-to-atom-part-1-of-4/>
- Programme pour développeurs AppUp : <http://appdeveloper.intel.com/fr-fr>
- MeeGo (en anglais) : <http://meego.com>
- Blogs du logiciel : <http://software.intel.com/fr-fr/blogs/>
- L'optimisation pour les Jeux (en anglais) : <http://software.intel.com/sites/billboard/va-magazine/issue-07/downloads/DevNetbooks-IntelVA7-July2010.pdf>
- QT : [http://qt.nokia.com/title-fr?set\\_language=fr&cl=fr](http://qt.nokia.com/title-fr?set_language=fr&cl=fr)



■ Jeff Horton

*Ingénieur d'application. Fort d'une expérience de vingt ans dans l'informatique, Il travaille chez Intel depuis plus de dix ans. Il est en charge actuellement de l'évangélisation des technologies Intel auprès de leaders dans le domaine de l'édition de logiciels. Il les assiste dans leurs développements et les guide dans leurs stratégies d'optimisations sur les plateformes à facteur de forme réduit basées sur Intel Atom.*

# 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)

# Economisez jusqu'à 50%



Programmez ! est le magazine du développement 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 ! [www.programmez.com](http://www.programmez.com)

**1** -25%

## Abonnement 1 an

**49€** au lieu de 65,45 € tarif au numéro - Tarif France métropolitaine

**2** +0,8€ par mois

## Abonnement Intégral : + archives

1 an au magazine + archives sur Internet et PDF  
**59€** Tarif France métropolitaine

**3** jusqu'à -50%

## Abonnement 2 ans + 1 livre numérique ENI

• **79€** au lieu de 130,90 (valeur de 22 numéros) Tarif France métropolitaine + un livre d'une valeur de 23,9 € à 31,9 €, soit un total de 154,8 € à 162,8 €

• **89€** 2 ans au magazine + archives sur Internet et PDF + 1 livre numérique ENI



### OUI, je m'abonne

*Vous pouvez vous abonner en ligne et trouver tous les tarifs [www.programmez.com](http://www.programmez.com)*

- ☐ Abonnement 1 an au magazine : **49 €** (au lieu de 65,45 € tarif au numéro) Tarif France métropolitaine
- ☐ Abonnement Intégral : 1 an au magazine + archives : **59 €** Tarif France métropolitaine
- ☐ Abonnement 2 ans au magazine + livre numérique ENI : **79 €** Tarif France métropolitaine
- ☐ Abonnement 2 ans au magazine + livre numérique ENI + archives : **89 €** Tarif France métropolitaine

**Livres à Choisir :** ☐ Visual Studio 2010 ☐ PHP5.3 ☐ Bing Maps ☐ MySQL 5, Administration et optimisation  
☐ Java et Spring, Concevoir, construire et développer une application Java/J2EE avec Spring. *Détails sur [www.programmez.com/abonnement.php](http://www.programmez.com/abonnement.php)*

☐ M. ☐ Mme ☐ Mlle Entreprise : \_\_\_\_\_ Fonction : \_\_\_\_\_  
Prénom : \_\_\_\_\_ Nom : \_\_\_\_\_  
Adresse : \_\_\_\_\_  
Code postal : \_\_\_\_\_ Ville : \_\_\_\_\_  
Tél : \_\_\_\_\_  
E-mail : \_\_\_\_\_ @ \_\_\_\_\_

(Attention, e-mail indispensable)

☐ 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  
28 février 2011

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.

# L'informatique booste la carrière des femmes

Microsoft et Ipsos ont réalisé une étude sur l'apport du numérique dans la réduction des inégalités professionnelles hommes/femmes. L'enquête a été menée auprès de 500 femmes françaises, actives ou en recherche d'emploi. Elle dresse 4 profils, en fonction de la relation aux nouvelles technologies. « Femmes numériques », « techno-demandeuses », ou « techno-défavorisées ».

## L'informatique accélère la carrière

40 % des femmes actives pensent que les technologies numériques pourraient améliorer leur situation professionnelle si elles les utilisaient davantage. Cette proportion passe à 51 % chez les femmes de 25-34 ans.

Si les technologies apparaissent comme des accélérateurs de carrière permettant d'évoluer, de se remettre à niveau, de télé-travailler, voire de créer sa propre entreprise, elles sont également perçues comme un levier majeur pour changer de métier ou retrouver un emploi après une interruption d'activité.

- 53 % des femmes en recherche d'emploi, désireuses de mieux profiter des nouvelles technologies, estiment que leur situation professionnelle pourrait s'améliorer grâce à elles.
- Les « techno-défavorisées », le plus souvent ouvrières, employées, résidant en milieu rural, de faible formation initiale, aspirent elles aussi à une vie

professionnelle numérique puisqu'elles sont même 62 % à souhaiter pouvoir profiter des technologies numériques.

Les « femmes numériques » (utilisatrices de l'ordinateur), se considèrent autant « disponibles » au travail que les hommes. Et ce d'autant plus qu'elles sont jeunes (51 % des moins de 35 ans) et bénéficient d'un niveau de formation élevée : 64 % des Bac+3 et plus.

Nathalie Wright, chargée du programme Diversité chez Microsoft France commente : « *Nous avons toujours eu la conviction que le numérique libérait les femmes, en permettant le travail de la maison, en horaire décalés, en rompant l'isolement, ou, plus prosaïquement, en permettant l'accès aux offres d'emplois. Cette étude corrobore nos intuitions et nous incite à l'action* ».

## Une demande de formation

61 % des femmes qui pensent que leur situation professionnelle pourrait s'amé-

liorer si elles utilisaient davantage les nouvelles technologies demandent de la formation afin de mieux savoir les utiliser. 60 % n'ont pas encore bénéficié d'une telle formation. Mais disposer d'un ordinateur à soi et d'une connexion Internet reste un privilège de « cols blancs ». Les femmes en situation précaire, en CDD et en intérim, et toutes celles qui occupent des emplois hors d'un bureau, sont exclues de ce système. Ainsi, seules 36 % des femmes en CDD ou en intérim, 20% des vendeuses et 26 % des ouvrières disposent d'une connexion Internet accessible pour leur travail.

Les femmes « techno-demandeuses » (femmes à la recherche d'un emploi, plus d'un tiers d'entre elles ont deux enfants ou plus) ressentent un besoin important de formation. Si 53 % d'entre elles pensent que leur situation professionnelle peut s'améliorer grâce aux technologies, 80 % des « techno demandeuses » expriment le besoin de savoir mieux les utiliser.

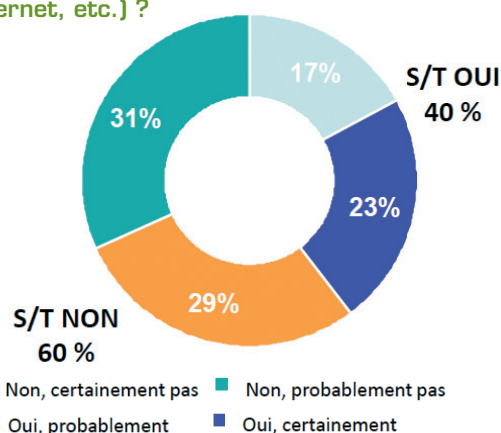
## Les Tics au service des femmes

Force Femmes accompagne et soutient les femmes de plus de 45 ans dans leurs démarches de retour à l'emploi et de création d'entreprise. De nombreuses femmes ont bénéficié de formations dispensées par les bénévoles de l'association au sein de la salle informatique aménagée par Microsoft.

Créée par Martin Hirsch en 2006, l'Agence nouvelle des solidarités actives (Ansa) lutte contre la pauvreté par la mise en place d'expérimentations

sociales. Depuis deux ans, Microsoft France est l'un des partenaires majeurs du programme TIC' Actives « le numérique au service de tous ». L'objectif : mettre les technologies de l'information au service de l'insertion et du retour à l'emploi. Microsoft fournit des ordinateurs reconditionnés et mobilise son réseau de partenaires, qui s'engagent en offrant équipements informatiques, prestations de services et de conseils, développement de sites Web, etc.

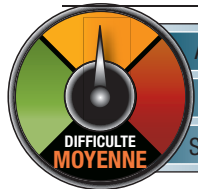
Pensez-vous que votre situation professionnelle pourrait s'améliorer si vous utilisiez davantage les nouvelles technologies numériques (informatique, Internet, etc.) ?





# Créer votre première application ASP.NET MVC 3 sur Windows Azure

Le but de cet article est de démontrer la simplicité avec laquelle nous pouvons créer des applications ASP.NET MVC 3 qui utilisent la plateforme Windows Azure.



APPLICATION : CLOUD

LANGAGE : ASP

SOURCE : OUI

La plateforme Windows Azure englobe l'offre complète de Microsoft sur le Cloud. Chaque service que Microsoft estime faire partie du Cloud sera proposé sous

cette étiquette. Le Cloud étant tout simplement un ensemble de serveurs qui hébergent et exécutent vos applications ou des services qui peuvent être consommés (comme les web services). Aujourd'hui, la plateforme Windows Azure est composée de Windows Azure, SQL Azure, Windows Azure AppFabric et Windows Azure Marketplace [Fig.1].

Passons maintenant à la pratique.

## VOTRE PREMIÈRE APPLICATION DANS LE CLOUD

Dans le cadre de cet exercice nous allons créer une simple application ASP.NET MVC 3 et l'héberger dans le Cloud. La bonne nouvelle est que pratiquement tout site web qui s'exécute sur IIS de Windows Server 2008 R2 peut être hébergé dans Windows Azure. Les applications suivantes sont des exemples de ce qu'Azure supporte :

- Les applications web ASP.NET 3.5 ou 4.0
- ASP.NET MVC 1.0, 2.0 ou 3.0 RC
- Les services web (WCF, ASMX).
- Les applications basées sur FastCGI comme PHP ou Python.
- Les applications Java et Ruby.

Avant de commencer à créer notre application nous avons besoin de Windows Azure Software Development Kit (SDK).

## PRÉPARATION DE NOTRE ENVIRONNEMENT DE DÉVELOPPEMENT

Au moment où j'ai terminé d'écrire cet article, la version la plus récente de SDK Windows Azure était la version 1.2 de Juin 2010. Entre temps la version 1.3 de SDK vient de sortir le 29/11/2010. Bien que le SDK 1.3 apporte beaucoup de nouvelles fonctionnalités, le processus de publication dans le cas de ASP.NET MVC 3 est exactement le même qu'avec la version 1.2. J'ai quand même effectué la mise à jour vers la nouvelle version pour vérifier que tout se passait parfaitement bien. Vous pouvez télécharger la dernière version de SDK sur [www.azure.com](http://www.azure.com). Le SDK contient plusieurs outils qui facilitent le développement des applications pour Windows Azure. Les outils suivants sont inclus :

- Windows Azure development fabric
- Les templates Visual Studio pour la création des applications web.
- L'environnement de stockage Windows Azure.
- Les divers outils de développement.

Avant d'installer le SDK, vérifiez votre version de Windows et de Visual Studio. Afin de pouvoir utiliser le SDK vous aurez également

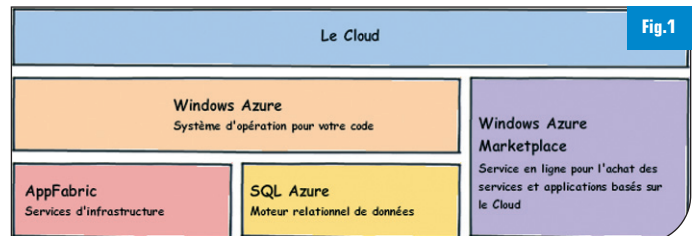


Fig.1

besoin d'une instance de SQL Server (Express ou normal).

Une deuxième étape consiste à télécharger et installer ASP.NET MVC 3 (à l'heure d'écriture de cet article la version la plus récente est MVC 3 RC). Vous pouvez la télécharger à l'adresse suivante : <http://www.microsoft.com/downloads/en/details.aspx?FamilyID=a920ccee-1397-4feb-824a-2dfefee47d54&displaylang=en>.

## LES SYSTÈMES D'EXPLOITATION SUPPORTÉS

Pour développer pour Windows Azure vous aurez besoin d'un de ces systèmes d'exploitation :

- Windows 7
- Windows Vista
- Windows Server 2008 et plus.

Windows XP n'est pas supporté par Windows Azure car les Web Roles de Windows Azure sont construits sur la base d'IIS 7.0. Windows XP et Windows Server 2003 utilisent les versions plus anciennes d'IIS non compatibles avec Windows Azure.

## LES VERSIONS DE VISUAL STUDIO SUPPORTÉES

Pour développer pour Windows Azure vous aurez besoin d'une de ces versions de Visual Studio :

- Visual Studio 2008
- Visual Studio 2010

Une fois installé le SDK, nous pouvons créer notre première application ASP.NET MVC 3 sur Azure.

**Note :** Afin d'exécuter une application Windows Azure dans la fabrique de développement à partir de Visual Studio, vous aurez besoin des privilèges d'administrateur. Pour cela cliquez avec le bouton droit de la souris sur l'icône de Visual Studio et choisissez l'option « Exécuter en tant qu'administrateur ».

## CRÉATION D'UN NOUVEAU PROJET

La première étape est la création d'un nouveau projet. Sélectionnez « File > New > Project ». Choisissez ensuite le template « Windows Azure Project » dans la section « Cloud » comme sur l'image de la [Fig.2]. Entrez un nom pour votre projet et cliquez sur « OK ». Une nouvelle fenêtre s'ouvre, dans laquelle vous devez choisir le type de projet Windows Azure que vous voulez créer. Les rôles suivants sont disponibles :

- ASP.NET Web Roles.
- ASP.NET MVC 2 Web Role
- WCF Service Web Role
- Worker Role
- CGI Web Role [Fig.3].

Comme vous pouvez le constater, il n'existe pas de rôle pour ASP.NET MVC 3, nous n'allons donc choisir aucun rôle. Nous devons suivre un autre chemin pour créer notre application ASP.NET MVC 3 prête à être déployée sur Azure. On clique sur le bouton « Ok » et on attend que Visual Studio nous génère la solution. La solution générée devrait ressembler à ceci : [Fig.4].

La prochaine étape consiste à ajouter un projet « ASP.NET MVC 3 » à la solution. Pour cela, faites un clic droit sur la solution et ajoutez un nouveau projet (« Add > New Project... ») : [Fig.5].

Dans la fenêtre qui s'ouvre nous allons alors sélectionner le template « ASP.NET MVC 3 Web Application ». Vous pouvez ensuite donner à ce projet le nom que vous souhaitez. Dans mon exemple j'ai choisi « ProgrammezMvc3WebApplication » : [Fig.6].

Cliquez ensuite sur le bouton « Ok ». Une nouvelle fenêtre s'affiche qui vous demande de configurer la création du projet « ASP.NET MVC 3 ». Dans le cadre de cet exemple nous allons sélectionner « Internet Application » et le moteur de vues « Razor ». J'ai également choisi de créer le projet de tests unitaires mais ce n'est pas nécessaire [Fig.7].

Vous validez en cliquant sur le bouton « Ok ». Votre solution devrait maintenant ressembler à ceci : [Fig.8].

L'étape existante consiste à ajouter un « Web Role » à notre service Cloud « ProgrammezCloudService ».

Pour le moment, dans Windows Azure deux rôles existent : Web Role et Worker Role. C'est tout simplement un autre nom pour votre application. Web role est une VM (machine virtuelle) qui héberge votre application dans IIS. Pour Worker Role, c'est la même chose que Web Role mais sans IIS. C'est prévu pour tous les traitements en tâche de fond.

Pour cela faites un clic droit sur le dossier « Role » et sélectionnez l'option « Add > Web Role Project in solution... » [Fig.9].

Dans la fenêtre qui s'affiche, sélectionnez notre projet ASP.NET MVC 3 « ProgrammezMvc3WebApplication » et cliquez sur « OK » [Fig.10].

Après avoir accepté en cliquant sur le bouton « OK », votre solution devrait ressembler à ceci : [Fig.11].

Avant d'exécuter notre application pour la première fois, nous devons d'abord comprendre les différences entre un « Web Role » et une application « ASP.NET MVC 3 Web Application ».

Il y a 4 différences principales. Un projet « Web Role » contient :

- Les références vers les assemblies de Windows Azure spécifiques : Microsoft.WindowsAzure.Diagnostics, Microsoft.WindowsAzure.ServiceRuntime et Microsoft.WindowsAzure.StorageClient.
- Le code d'initialisation (Bootstraper) qui habituellement se trouve dans le fichier WebRole.cs/vb. Ce code démarre DiagnosticMonitor et définit également le comportement par défaut de recyclage de rôle quand une modification dans la configuration intervient.
- Une configuration de trace listener dans le fichier Web.config. Microsoft.WindowsAzure.Diagnostics.DiagnosticMonitorTraceListener.
- Dans le cas des applications web MVC 3 avec le moteur des vues Razor, les assemblies suivantes doivent être présentes :

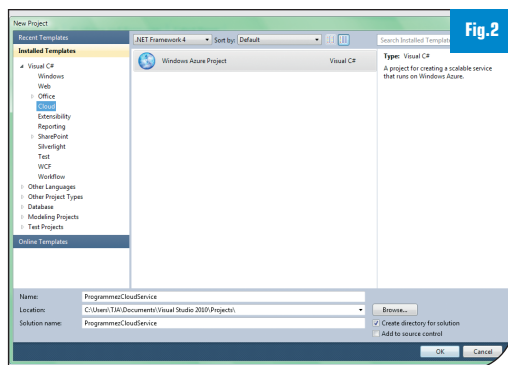


Fig.2

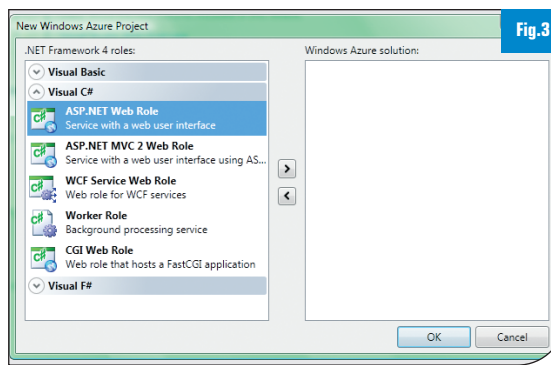


Fig.3

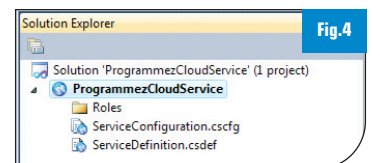


Fig.4

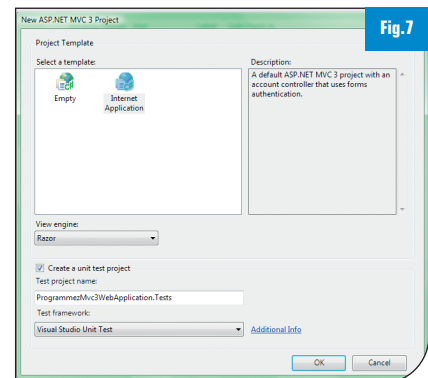


Fig.7

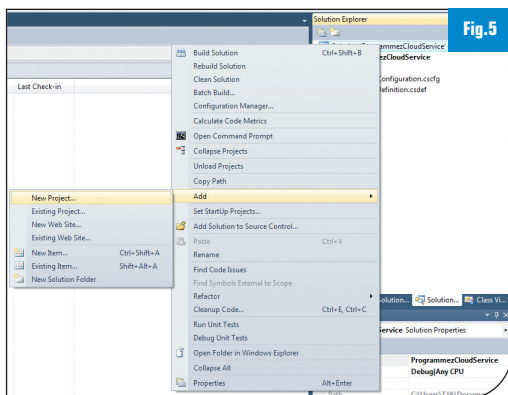


Fig.5

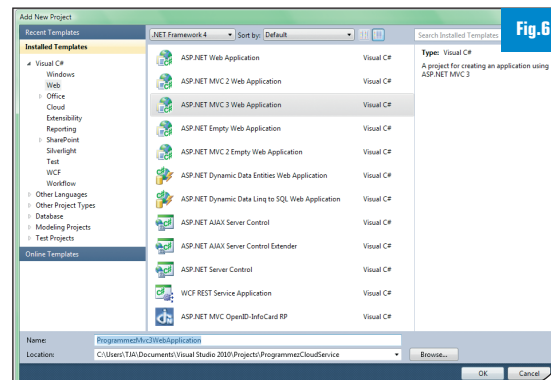


Fig.6

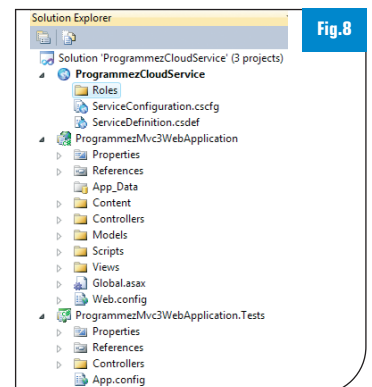


Fig.8

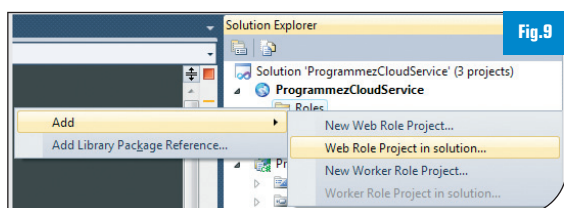


Fig.9

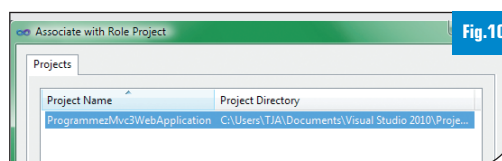


Fig.10

- System.Web.Mvc
- Microsoft.Web.Infrastructure
- System.Web.Razor
- System.Web.WebPages
- System.Web.WebPages.Razor
- System.Web.Helpers

Toutes ces références doivent avoir la propriété « Copy Local » positionnée à la valeur « True ». Ceci nous assure que ces assemblies seront disponibles dans le Cloud. Les VM de Cloud contiennent uniquement les assemblies du Framework.NET 3.5 SP1 et Framework.NET 4.0. Les assemblies pour ASP.NET MVC 3 n'en font pas partie. Le fait de positionner la propriété « Copy Local » à « True » assurera que ces assemblies seront incluses au Service Package qui sera chargé dans le Cloud pour exécuter notre application.

Puisque nous n'avons pas tout à fait poursuivi le chemin standard de création de projet prêt à être déployé sur Azure (souvenez-vous, le template de création pour ASP.NET MVC 3 n'existe pas), nous devons ajouter ces prérequis manuellement.

1. Ajouter les références vers les assemblies Microsoft.WindowsAzure.Diagnostics, Microsoft.WindowsAzure.ServiceRuntime et Microsoft.WindowsAzure.StorageClient. Chemin d'installation par défaut est : « C:\Program Files\Windows Azure SDK\v1.3\ref\ ». Les assemblies Microsoft.WindowsAzure.Diagnostics et Microsoft.WindowsAzure.StorageClient doivent avoir également positionné la propriété « Copy Local » à la valeur « True ».
2. Ajouter le fichier WebRole.cs (ou .vb pour le site web en VB.NET) au projet ASP.NET MVC 3 « ProgrammezMvc3WebApplication ». Le code ressemble à ceci (version C#) :

```
using System.Linq;
using Microsoft.WindowsAzure.Diagnostics;
using Microsoft.WindowsAzure.ServiceRuntime;

namespace ProgrammezMvc3WebApplication
{
    public class WebRole : RoleEntryPoint
    {
        public override bool OnStart()
        {
            DiagnosticMonitor.Start("DiagnosticsConnectionString");

            // For information on handling configuration changes
            // see the MSDN topic at http://go.microsoft.com/fwlink/?LinkId=166357.
            RoleEnvironment.Changing += RoleEnvironmentChanging;

            return base.OnStart();
        }

        private void RoleEnvironmentChanging(object sender, RoleEnvironmentChangingEventArgs e)
        {
            // If a configuration setting is changing
            if (e.Changes.Any(change => change is RoleEnvironmentConfigurationSettingChange))
            {
                // Set e.Cancel to true to restart this role instance
                e.Cancel = true;
            }
        }
    }
}
```

```
}
}
}
}
```

3. Ajouter la configuration de trace listener dans le fichier Web.config de notre projet ASP.NET MVC 3 :

```
<system.diagnostics>
  <trace>
    <listeners>
      <add
        type="Microsoft.WindowsAzure.Diagnostics.DiagnosticMonitorTrace
        Listener, Microsoft.WindowsAzure.Diagnostics, Version=1.0.0.0,
        Culture=neutral, PublicKeyToken=31bf3856ad364e35"
        name="AzureDiagnostics">
      </add>
    </listeners>
  </trace>
</system.diagnostics>
```

4. Positionner la propriété « Copy Local » à « True » des références suivantes :

- System.Web.Mvc
- System.Web.Helpers
- System.Web.WebPages [Fig.12]

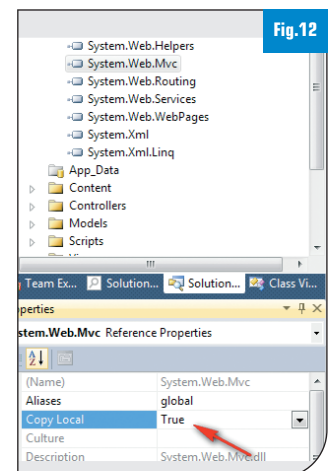
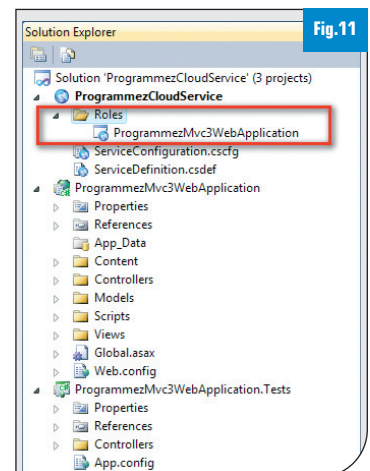
Ceci n'est pas cependant suffisant. Il faut également que le package que nous allons déployer sur Azure contienne les assemblies suivantes :

- Microsoft.Web.Infrastructure
- System.Web.Razor
- System.Web.WebPages.Razor

Bien qu'elles ne soient pas nécessaires en tant que références du projet, nous devons les ajouter afin qu'elles soient déployées sur Windows Azure. Elles se trouvent par défaut dans le répertoire « C:\Program Files\Microsoft ASP.NET\ASP.NET Web Pages \v1.0\Assemblies ». N'oubliez pas de positionner également la propriété « Copy Local » à « True » Nous sommes maintenant prêts à exécuter notre application.

## EXÉCUTION DE L'APPLICATION WEB

Avant d'appuyer sur « F5 » pour exécuter notre « Web Role » nous devons nous assurer que notre service Cloud « ProgrammezCloudService » est sélectionné en tant que projet de démarrage au





lieu de notre application ASP.NET MVC 3. Le but est de faire exécuter notre application à l'aide de Azure SDK. Maintenant c'est la partie la plus excitante. Nous sommes enfin prêts à exécuter notre application au sein de la fabrique de développement de Windows Azure.

Appuyez sur « F5 » comme dans le cas d'autres applications Visual Studio. Si vous exécutez pour la première fois votre application dans la fabrique Windows Azure, il faut d'abord configurer la fabrique.

**Note :** Par défaut quand la fabrique de développement Windows Azure est configurée et plus précisément Windows Azure Development Storage, l'instance de SQL Server 'localhost\SqLExpress' est recherchée. Si votre instance porte un autre nom, vous devez reconfigurer Development Storage à l'aide de l'outil 'DSInit' qui se trouve dans Windows Azure SDK.

Si la configuration se passe bien vous devriez avoir une fenêtre qui ressemble à ceci : [Fig.13].

Vous cliquez sur « OK » et vous verrez une petite icône bleue dans la barre des tâches de Windows, ce qui signifie que votre application s'exécute au sein de la fabrique Windows Azure : [Fig.14].

Ensuite, la fenêtre de notre application s'affiche, comme dans le cas d'autres applications web : [Fig.15].

Félicitations ! Vous avez développé votre première application ASP.NET MVC 3 au sein de Windows Azure ! La prochaine étape consiste à déployer notre application dans les serveurs de production de Windows Azure.

## DÉPLOIEMENT DANS WINDOWS AZURE LIVE

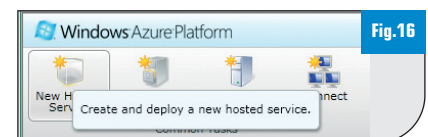
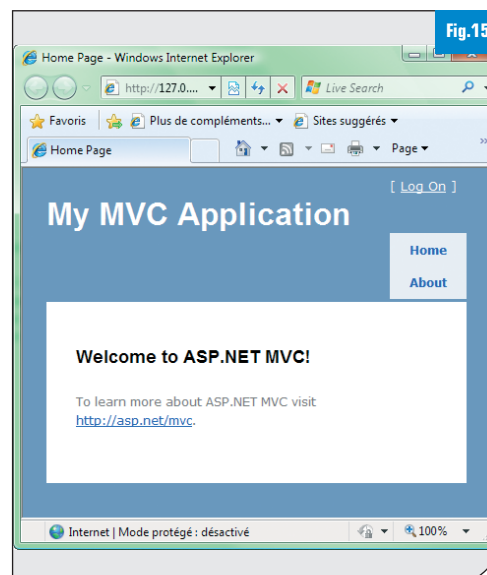
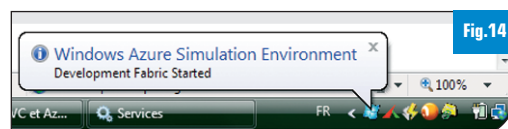
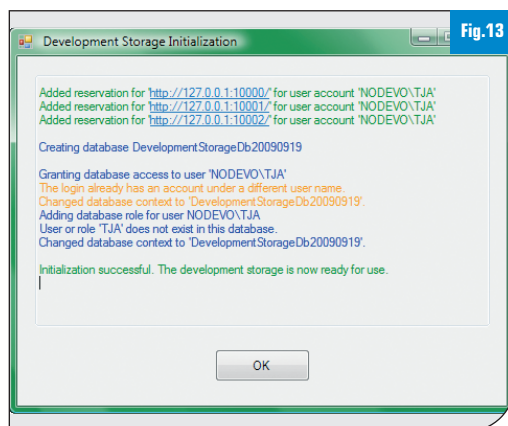
Avant de pouvoir déployer votre application, vous devez disposer d'un compte Windows Azure. Pour en ouvrir un, veuillez-vous rendre à l'adresse suivante : <http://www.microsoft.com/windowsazure/offers/>. Une fois votre choix fait, suivez les étapes de création de compte. Quand vous aurez terminé, vous pouvez vous loguer avec votre Live ID sur <http://www.azure.com> pour accéder à la liste de vos projets. Le portail Azure est le point central de gestion de tous les aspects de vos applications qui s'exécutent sur Windows Azure.

**Note :** Depuis la sortie du SDK 1.3 fin novembre, un nouveau portail Silverlight Windows Azure a vu le jour. Bien que l'ancien soit encore en fonction, il vaut mieux s'habituer à utiliser le nouveau car ce premier risque de disparaître début 2011.

Dans l'abonnement Windows Azure dont je dispose j'ai déjà un projet intitulé « CvMedia ». Maintenant nous devons créer un nouveau service en cliquant sur l'icône « New Hosted Service » se trouvant dans la barre des tâches : [Fig.16].

Une fenêtre pop-up de création s'ouvre. Dans cette fenêtre, vous devez tout d'abord choisir un projet auquel vous voulez lier votre nouveau service. J'ai choisi celui qui existe dans mon abonnement « CvMedia ». Vous devez ensuite donner un nom et un préfixe d'Url pour votre service. J'ai choisi « aspnetmvc3 ». Vous pouvez également renseigner les groupes d'affinité. Cela permet de déployer les services associés au même centre régional de données pour une question de performances ou pour les centres différents au cas d'un incident. Nous pouvons également indiquer les options de déploiement. Par défaut on déploie dans l'environnement de test « Staging » afin que vous puissiez vous assurer que votre application se comporte correctement dans le Cloud. Le processus typique de déploiement d'applications consiste à déployer tout d'abord le projet dans l'environnement « Staging », le tester et le « basculer » dans l'environnement de production. Le basculement s'appelle VIP swap. VIP signifie Virtual IP. Déployer le projet directement dans l'environnement de production est risqué si votre application présente des bugs non détectés lors du développement. Vous pouvez en profiter pour charger votre application en spécifiant le nom de déploiement « Deployment name », en choisissant l'emplacement de package « Package location » et le fichier de configuration « Configuration file ». Nous n'allons pas le faire dans cette étape. Voici à quoi ressemble notre écran : [Fig.17].

La partie la plus importante est le nom public que vous allez donner à votre service. Ce nom sera associé à tout ce que vous faites avec votre service, donc choisissez-le avec beaucoup d'attention. Ce nom constituera également une partie de l'URL par laquelle on pourra accéder à votre service. Dans notre cas ce sera : <http://aspnetmvc3.cloudapp.net>. Il ne nous reste plus qu'à cliquer sur le bouton « OK » et à retourner sur la page principale : [Fig.18]. La deuxième étape consiste à créer un autre service « Storage Account » en cliquant sur l'icône « New Storage Account » : [Fig.19]. Dans la fenêtre qui s'ouvre il suffit de choisir le projet Azure existant (dans mon cas « CvMedia ») et de spécifier un nom (j'ai choisi « cvmedia » également). Vous pouvez aussi spécifier les groupes d'affinité pour le stockage de vos données : [Fig.20].



## PACKAGER ET DÉPLOYER NOTRE APPLICATION

Avant de déployer notre application, nous devons modifier un paramétrage dans le fichier « ServiceConfiguration.cscfg » de notre projet « ProgrammezCloudService ». En fait, par défaut, le service est configuré pour utiliser le service de stockage local.

```
<Setting name="DiagnosticsConnectionString" value="UseDevelopmentStorage=true" />
```

Le service de stockage local qui se trouve sur votre machine n'est pas accessible à partir de Azure. Puisque nous avons créé un service de stockage, nous allons modifier ce paramètre. La nouvelle valeur est la suivante :

```
<Setting name="DiagnosticsConnectionString" value="DefaultEndpointsProtocol=https;AccountName=[VOTRE NOM DE COMPTE];AccountKey=[VOTRE CLÉ DE COMPTE]" />
```

La clé et le nom de votre compte sont disponibles dans la partie « Storage Accounts » sur le portail Windows Azure : [Fig.21].

Nous sommes enfin prêts à déployer notre application. Pour cela nous allons créer d'abord un package. Une des manières de le faire serait d'employer l'outil « CSPack » de Windows Azure SDK, mais puisqu'on utilise Visual Studio, il suffit de faire un clic droit sur le projet Cloud (dans notre cas « ProgrammezCloudService ») et de choisir l'option « Publish ». Cette commande appelle « CSPack » pour vous en lui passant tous les paramètres nécessaires. Dans Visual Studio la fenêtre de publication de service Cloud s'ouvre : [Fig.22].

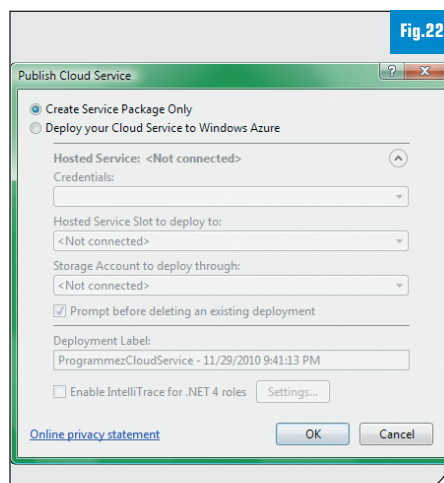
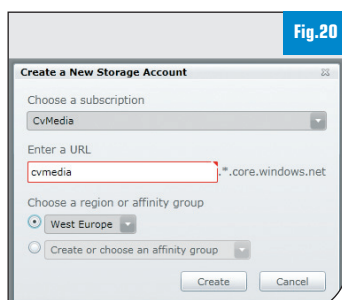
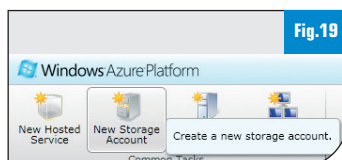
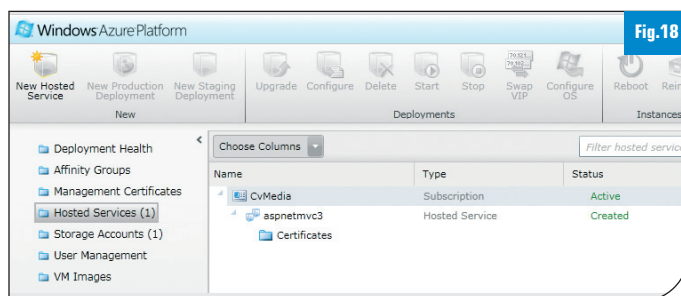
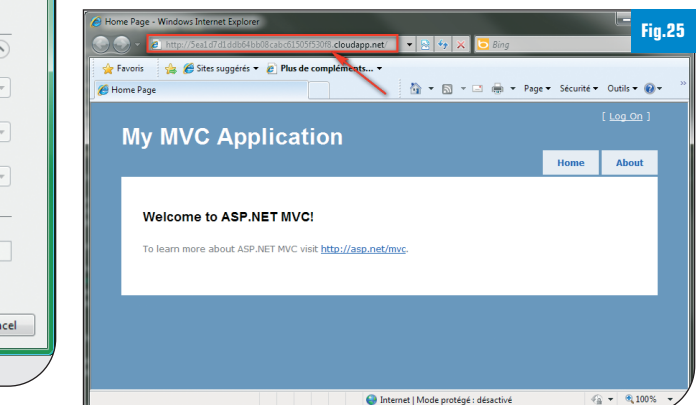
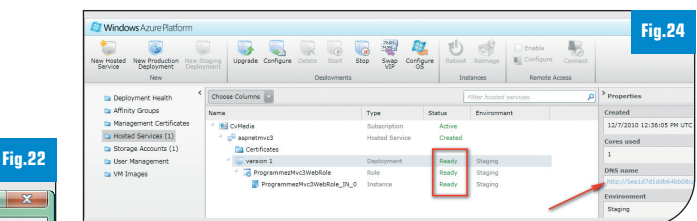
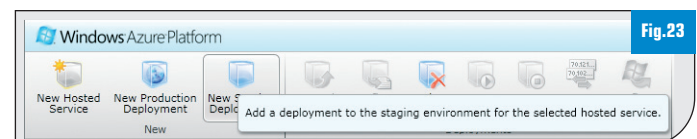
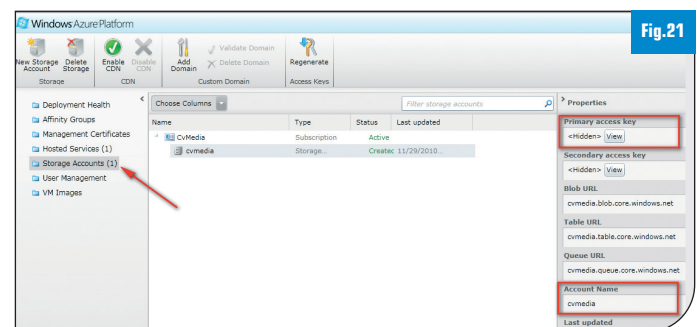
Par défaut, l'option « Deploy your Cloud Service to Windows Azure » est sélectionnée. Vous pouvez vous connecter grâce à un certificat et déployer votre application automatiquement. Nous allons cepen-

tant choisir l'option « Create Service Package Only » et déployer notre application manuellement via le portail Azure. Nous cliquons donc sur le bouton « Ok » et attendons que Visual Studio nous génère le package. Une fois la génération terminée, la fenêtre Windows Explorer s'ouvre à l'emplacement où le package a été créé.

Nous devons ensuite retourner sur le portail Azure à la page « Hosted Services ». Puis, vous cliquerez sur l'icône « New Staging Deployment » : [Fig.23]. Dans la fenêtre qui s'ouvre, nous allons choisir le package et le fichier de configuration générés dans l'étape précédente par Visual Studio. Ensuite, il suffira de donner un nom lié à notre déploiement comme par exemple « version 1 » et nous cliquerons sur le bouton « OK ». Notre package est déployé sur la plateforme Windows Azure et si le processus de déploiement se passe correctement vous devriez voir le statut « Ready » à côté de votre rôle comme sur l'écran ci-dessous : [Fig.24].

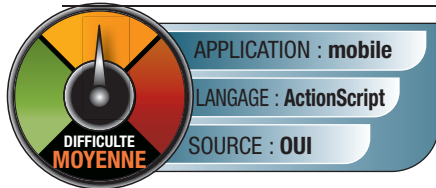
Pour tester notre déploiement, il suffit d'ouvrir une fenêtre de navigateur internet et de taper l'url spécifiée dans la propriété « DSN name » du portail Windows Azure. Vous verrez ensuite la page d'accueil de notre application s'exécutant sur Azure !

■ Thomas Jaskula - NODEVO  
Associé, Chef de Projet .NET



# Combinaison Flex et Android c'est possible !

Cet article a pour but de vous expliquer comment créer une application pour téléphone mobile équipé d'Android à l'aide de Flash Burrito, du framework Flex 4.5 et du runtime Adobe Air 2.5. Nous verrons comment créer un projet à l'aide de Flash Burrito, puis nous le déploierons sur un téléphone émulé par le SDK Android. Afin d'aborder simplement la création d'applications Flex pour Android, j'ai choisi de vous présenter le développement d'un métronome qui pourra sans doute être utile à tout musicien.



Le framework Flex s'est vu modifié vers la fin de l'été 2010 pour pouvoir accueillir de nouvelles fonctionnalités dédiées au monde mobile. Une version

4.5 est alors mise en ligne ayant pour nom de code « Hero ». Pour exécuter une application développée avec le framework Flex sur un environnement bureautique, il est nécessaire d'utiliser le runtime Adobe Air. Ce runtime a alors lui aussi évolué pour pouvoir prendre en compte les aspects mobilité utilisés par le framework 4.5 et pouvoir s'exécuter sur les plateformes mobiles.

## CRÉATION DU PROJET

Pour créer le projet, il faut être en possession de l'environnement de développement Flash Burrito, disponible en téléchargement sur le site d'Adobe. A noter que cette version est en libre essai pendant 60 jours, mais si vous possédez déjà une clé valide pour Flash Builder celle-ci validera également le nouvel environnement de développement. Nous allons commencer par créer un nouveau projet de type « Projet Flex mobile » via le menu fichier>nouveau de Flash Burrito. S'exécute alors un wizard dont la première fenêtre nous propose de saisir le nom du projet (que nous nommerons : « mobMetronome »). La fenêtre suivante nous offre des options concernant la plateforme cible (Android), puis d'utiliser ou non un modèle d'application. Enfin, nous pouvons choisir le mode d'exécution de l'application : mode plein écran ou non [Fig.1].

La troisième fenêtre, nous propose de sélectionner une technologie

serveur (PHP, JEE...) mais comme nous n'allons pas utiliser ces technologies, nous laissons les options sélectionnées par défaut. Enfin, nous terminons en validant la configuration des chemins de compilations présentés dans le dernier écran du wizard. Notre projet est à présent créé.

## Choix d'un émulateur

La première option qui s'offre à nous est d'utiliser un émulateur proposé par l'environnement de développement. Pour cela, il faut exécuter l'application en mode débogage afin de voir apparaître un écran de configuration. Dans cet écran, en sélectionnant l'option de méthode de lancement « Sur le bureau » nous avons la possibilité de choisir le type d'unité mobile à émuler. Etant donné que nous allons exécuter l'application sur un émulateur Android ayant des caractéristiques spécifiques nous allons créer notre propre émulateur Flex : Cliquer sur le bouton configurer, puis saisir les valeurs présentes dans la [Fig.2]. Une fois l'émulateur créé, il faut le sélectionner dans la liste déroulante présente sur l'écran de paramétrage du mode de débogage comme l'illustre la [Fig.3] :

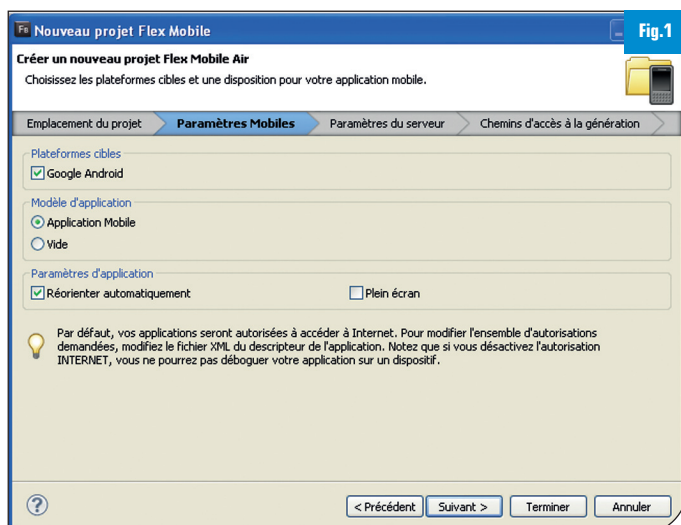
## Utilisation de son propre appareil mobile

Une autre option consiste à tester l'application directement sur son propre téléphone mobile. Dans ce cas, il suffit de sélectionner l'option « Sur le dispositif » et l'appareil est automatiquement détecté à condition que celui-ci soit connecté à l'ordinateur via un port USB.

## DÉVELOPPEMENT DE L'INTERFACE GRAPHIQUE

Nous allons à présent créer l'interface graphique de notre application. Mais avant d'aller plus loin, je vous invite à ouvrir le fichier mobMetronome.mxml situé dans le package par défaut.

Pour une application Flex standard, c'est dans ce fichier que nous devons placer nos composants. Nous allons voir que cela est un peu différent pour une application mobile.



Paramétrage du projet

```
<?xml version="1.0" encoding="utf-8"?>
<s:MobileApplication xmlns:fx="http://ns.adobe.com/mxml/2009"
    xmlns:s="library://ns.adobe.com/flex/spark" first
View="views.mobMetronomeAccueil">
  <fx:Declarations>
    <!-- Placer ici les éléments non visuels (services et objets
de valeur, par exemple). -->
  </fx:Declarations>
</s:MobileApplication>
```

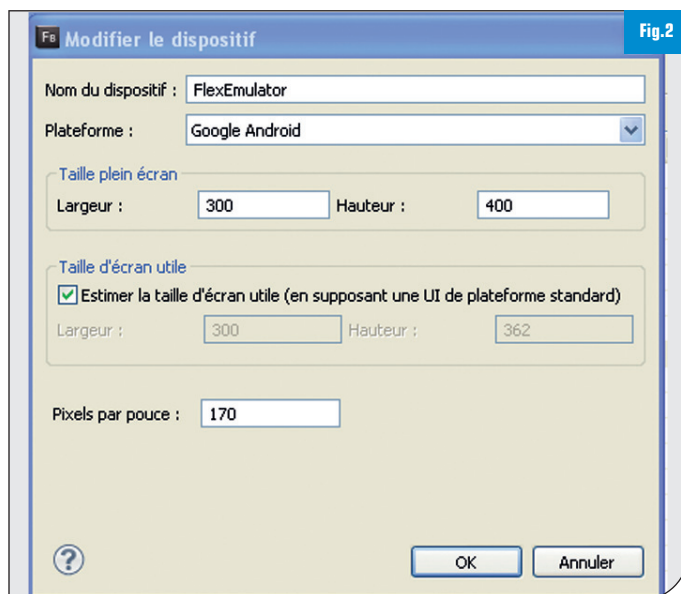
A la lecture de ces lignes de code, nous pouvons observer que la



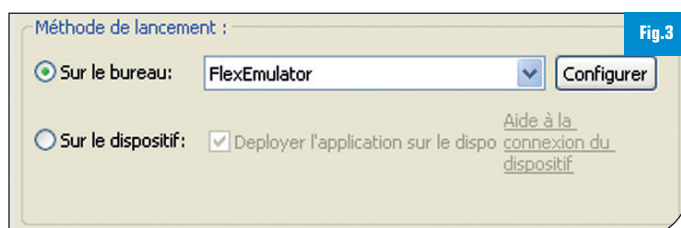
librairie de composants Spark s'est agrandie avec l'apparition d'une nouvelle classe (ou composant) `MobileApplication` qui, comme son nom l'indique, est dédiée à la création d'applications mobiles. Une autre chose remarquable est l'utilisation de l'attribut `first-view` de la classe `MobileApplication` nous permettant d'indiquer la première « vue » de notre application qui se trouve dans le package `views` et portant le nom de `mobMetronomeAccueil`. Si nous nous déplaçons dans ce package, nous pouvons voir l'existence d'un fichier de description MXML. C'est donc le fichier `mobMetronomeAccueil.mxml` qu'il faut éditer afin de créer l'interface graphique de l'application. Comme vous pouvez le constater en passant dans la vue création de Flash Burrito, nous n'avons que peu de composants à notre disposition. Mais ceux-ci suffiront à la création de notre projet [Fig.4]. Voici donc le code à implémenter :

```
<s:View xmlns:fx="http://ns.adobe.com/mxml/2009"
  xmlns:s="library://ns.adobe.com/flex/spark" title="Metronome"
  xmlns:local="*" creationComplete="enabledIHM()">
  <fx:Declarations>
    <!-- Placer ici les éléments non visuels (services et objets
  de valeur, par exemple). -->
  </fx:Declarations>

  <
  <s:Button id="btn_start" x="195" y="18" height="48" label="
  Start" click="play()" />
  <s:Button id="btn_stop" x="195" y="74" height="48" label="
  Stop" click="stop()" />
```



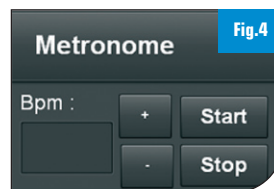
Création d'un dispositif



Utilisation de l'émulateur

```
<s:TextInput id="txt_bpm" x="10" y="47" width="108" height="
  62" fontSize="32" text="" textAlign="center" />
  <s:Button id="btn_plus" x="124" y="16" width="60" height="
  50" label="+" click="plusBpm()"
    fontSize="14" />
  <s:Button id="btn_minus" x="126" y="74" width="60" height="
  50" label="-" click="minusBpm()"
    fontSize="14" />
  <s:Label x="10" y="16" text="Bpm :"/>

</s:View>
```



Interface graphique

## LA CLASSE MÉTRONOME

Nous n'allons pas entrer dans le détail de cette classe car de nombreux commentaires s'y trouvent. A noter cependant l'utilisation d'un `dispatchEvent` permettant de notifier chaque changement de temps du métronome. A chaque changement de temps (de 1 à 4 correspondant à une mesure à 4 temps), la méthode `getBeat()` est exécutée. Via la notion de `databinding`, nous pouvons alors lier un composant de notre interface graphique à cette méthode afin de connaître le temps en cours et l'afficher pour information. Nous utilisons deux sons différents :

- Un pour le tempo (`_soundTempo`)
- Un autre pour indiquer que nous sommes au premier temps de la mesure. (`_soundFirstBeat`)

Le temps écoulé entre chaque battement de mesure est réalisé par un timer dont l'intervalle est calculé comme suit :

$$1000 / (\text{Battement par minute} / 60)$$

Une seconde = 1000 millisecondes. Donc, pour un tempo à 120 battements par minutes, nous avons 2 battements par secondes, soit un intervalle de 500 millisecondes entre chaque son.

```
package com.aurelienvannieuwenhuyze.metronome
```

```
{
  import flash.events.Event;
  import flash.events.EventDispatcher;
  import flash.events.TimerEvent;
  import flash.media.Sound;
  import flash.net.URLRequest;
  import flash.utils.Timer;

  [Bindable] //Utile pour réaliser le databinding
  public class Metronome
  {
    private var _soundTempo:Sound;
    private var _soundFirstBeat:Sound;
    private var _bpm:int;
    private var _mesure:int = 4; //Mesure à 4 temps
    private var _beat:int;
```

```

private var _timer:Timer;

public function Metronome()
{
    this._soundTempo=new Sound();
    this._soundTempo.load(new URLRequest('click.mp3'));
    this._soundFirstBeat=new Sound();
    this._soundFirstBeat.load(new URLRequest('beep-7.mp3'));
}

/*****
 *      COMMANDES      *
 * *****/

public function start():void
{
    this.playSoundTempo();
    this.playFirstBeat();
    _beat = 1;
    _timer = new Timer(int(1000/(_bpm/60)));
    _timer.start();
    _timer.addEventListener(TimerEvent.TIMER, playSequence)
}

public function stop():void
{
    _timer.stop();
}

/*****
 *      LA SEQUENCE DE BATTEMENTS      *
 * *****/

private function playSequence(e:TimerEvent):void
{
    _beat++;
    if (this._beat > this._mesure)
    {
        _beat=1;
        _soundFirstBeat.play();
    }
    _soundTempo.play();

    // DispatchEvent: Déclenchement de la methode get Beat().
    dispatchEvent(new Event(«beatChanged»));
}

//Renvoie le temps actuel de la mesure
[Bindable(«beatChanged»)]
public function get Beat():int

```

```

{
    return _beat;
}

/*****
 *      GETTER SETTER Et SONS      *
 * *****/

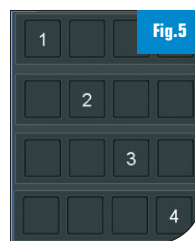
public function set Bpm(bpm:int):void
{
    _bpm=bpm;
}

public function playSoundTempo():void
{
    _soundTempo.play(0);
}

public function playFirstBeat():void
{
    _soundFirstBeat.play(0);
}

}

```



Composant d'affichage des temps

## CRÉATION D'UN COMPOSANT D'AFFICHAGE DES TEMPS DE LA MESURE

Afin d'indiquer à l'utilisateur le temps de la mesure correspondant au son entendu, nous allons créer un composant prenant en paramètre le numéro du temps de la mesure renvoyé par la méthode `get Beat()` de la classe métronome. En fonction du numéro [1 à 4],

une zone de texte sera activée [Fig.5] :

Pour créer un composant (que nous nommerons `cpnBeatDisplay`), réaliser un clic droit sur le nom du projet, puis sélectionner l'option nouveau > Composant MXML. Notre composant doit implémenter le composant `spark.components.BorderContainer` afin de pouvoir disposer les zones de texte permettant d'afficher le temps de la mesure en cours. Pour cela, saisir `spark.components.BorderContainer` dans la zone de texte « Basé sur ». Une fois le composant créé, saisir ces quelques lignes permettant d'implémenter le comportement décrit précédemment :

```

<?xml version="1.0" encoding="utf-8"?>
<s:BorderContainer xmlns:fx="http://ns.adobe.com/mxml/2009"
    xmlns:s="library://ns.adobe.com/flex/spark"
    width="236" height="66" backgroundColor="#3F3F3F">
<fx:Declarations>
    <!-- Placer ici les éléments non visuels (services et objets

```

```

de valeur, par exemple). ->
</fx:Declarations>

<fx:Metadata>
    [Bindable]
</fx:Metadata>

<fx:Script>
    <![CDATA[

        import flash.events.Event;

        private var _beat:int;

        public function set beat(value:int):void
        {
            _beat = value;
            Display()
        }

        public function Display():void
        {
            switch(_beat)
            {
                case 1:
                    txt_beat_1.text="1";
                    txt_beat_2.text="»";
                    txt_beat_3.text="»";
                    txt_beat_4.text="»";
                    break;

                case 2:
                    txt_beat_1.text="»";
                    txt_beat_2.text="2";
                    txt_beat_3.text="»";
                    txt_beat_4.text="»";
                    break;

                case 3:
                    txt_beat_1.text="»";
                    txt_beat_2.text="»";
                    txt_beat_3.text="3";
                    txt_beat_4.text="»";
                    break;

                case 4:
                    txt_beat_1.text="»";
                    txt_beat_2.text="»";
                    txt_beat_3.text="»";
                    txt_beat_4.text="4";
                    break;
            }
        }
    ]]>

```

```

</fx:Script>

<s:TextArea id="txt_beat_1" x="10" y="6" width="49" height="
51" editable="false"
            enabled="true" text="1" textAlign="center" vertical
Align="middle"/>
<s:TextArea id="txt_beat_2" x="67" y="6" width="49" height="
51" editable="false"
            textAlign="center" verticalAlign="middle"/>
<s:TextArea id="txt_beat_3" x="124" y="6" width="49" height
="51" editable="false"
            textAlign="center"/>
<s:TextArea id="txt_beat_4" x="181" y="6" width="49" height
="51" borderVisible="true"
            contentBackgroundColor="#3F3F3F" editable="false" enabled
="true" textAlign="center"/>
</s:BorderContainer>

```

## UTILISATION DU COMPOSANT DANS LE PROJET

Notre composant créé, nous pouvons à présent l'incorporer dans notre application. Nous pouvons également ajouter les actions correspondant à chaque bouton de l'interface sans oublier le databinding que nous avons mis en place dans la classe Métronome appliquée au composant nouvellement créé : voir code source complet sur notre site.

Nous pouvons à présent tester notre projet et constater son bon fonctionnement [Fig.6].

## EMULATION SUR ANDROID

Afin de vérifier son bon fonctionnement, nous avons la possibilité de la déployer sur un téléphone mobile ou sur un téléphone émulé via le SDK d'Android. Cette dernière solution restant la moins onéreuse.

### Prérequis

Afin de réaliser ce qui va suivre, il est nécessaire que le SDK d'Android soit correctement installé et configuré sur votre environnement. Pour cela, je vous invite à consulter le site d'Android à cette adresse : <http://www.android.com/>. Dans mon cas, j'ai installé le SDK d'Android dans le répertoire suivant : C:\ANDROID. J'ai également créé une variable d'environnement ANDROID\_SDK\_HOME pointant vers ce répertoire. Enfin, ne pas oublier d'exécuter le serveur Android via la ligne de commande suivante :

```

cd c:\ANDROID\Tools
adb start-server

```

Un autre prérequis est le téléchargement du SDK d'Adobe Air, disponible sur le site d'Adobe. Une fois le téléchargement effectué, il faut décompresser l'archive à un endroit à votre convenance (C:\AIR\_SDK pour ma part)

## Création d'un Mobile dans le SDK Android

Pour créer un téléphone mobile virtuel, nous pouvons utiliser SDK Manager. Cependant pour avoir plus d'options il est préférable de créer le téléphone en ligne de commande



```
cd c:\ANDROID\Tools
android create avd -n flexdevice -t 7 -p c:/ANDROID/Devices/
FlexDevice
```

-t correspondant à l'identifiant de la version d'OS sélectionnée dans la fenêtre du SDK Manager lors de la création du téléphone [Fig.7].

-p au répertoire de destination des fichiers de configuration du téléphone virtuel.

-n Le nom du téléphone virtuel

Lorsque cette commande est exécutée, vous avez la possibilité de choisir les options telles que l'utilisation de la caméra, de la capacité de mémoire... Une fois l'appareil créé, nous pouvons lancer son émulation à l'aide du SDK Manager [Fig.8].

## Installation d'Adobe Air 2.5 sur le téléphone

Avant de tester notre application sur notre tout nouveau téléphone virtuel, nous devons installer le runtime Adobe Air 2.5. Ce dernier se trouve dans le répertoire C:\AIR\_SDK\runtimes\air\android\emulator du SDK d'Adobe Air. Pour plus de facilité lors des étapes suivantes, placer le runtime dans un répertoire tel que C:\ANDROID\AIR, puis ouvrir ensuite une console Windows. Enfin, saisir la commande suivante :

```
Cd c:\ANDROID\Tools
adb install C:\ANDROID\AIR\Runtime.apk
```

### Important : Version du runtime

Il faut utiliser le runtime dédié à être utilisé sur un émulateur C:\AIR\_SDK\runtimes\air\android\emulator

Si tout se déroule correctement, Adobe Air s'installe sur le téléphone virtuel (Ne pas oublier d'activer le téléphone virtuel) :

```
C:\ANDROID\tools>adb install C:\ANDROID\AIR\Runtime.apk
1057 KB/s (0 bytes in 6784204.006s)
pkg: /data/local/tmp/Runtime.apk
Success
```

## Exécution du projet Flex sur l'appareil émulé

Flash Burrito possède un packager d'application capable de générer l'application avec l'extension .apk. Cependant, cette façon de procéder ne semble pas être valide pour l'utilisation sur un émulateur car l'application générée ne prend pas en compte le fait d'être exécutée sur un émulateur. Il faudra donc réaliser ce packaging en ligne de commande à l'aide du SDK AIR 2.5

### Création d'un certificat

Pour qu'une application soit installable sur un appareil mobile, il faut que celle-ci soit protégée par un certificat. Flash Burrito réalise correctement cette étape, nous n'allons donc pas nous priver de ses services. Réaliser un clic droit sur le nom du projet Flex, sélectionner l'option « exporter » puis choisir l'item « version validée » dans le menu Flash Builder. Dans la première fenêtre, **décocher la case** « Deploy application to any connected devices », puis valider.

L'écran suivant propose la création d'un certificat. Je vous laisse le soin de saisir le répertoire et le mot de passe que vous souhaitez puis de terminer la génération de l'application.

### Packaging de l'application

Nous allons à présent procéder au packaging de l'application afin de pouvoir ensuite la déployer sur notre appareil virtuel.

Nous allons copier les fichiers suivants dans le répertoire bin du SDK Adobe Air :

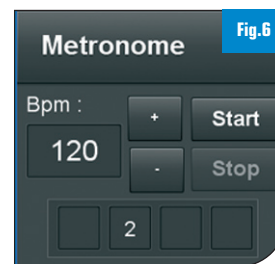
- Le certificat généré précédemment
- Le fichier mobMetronome-app.xml situé dans le répertoire bin-debug du projet Flex
- Le fichier mobMetronome.swf situé également dans le répertoire bin-debug du projet Flex
- Les deux fichiers MP3 : click.mp3 et beep-7.mp3

Ouvrons à présent une invite de commande et plaçons-nous dans le répertoire bin du SDK Adobe Air pour saisir la commande suivante ou nous spécifions la création d'une application pour un émulateur [apk-emulator] :

```
C:\AIR_SDK\bin>adt -package -target apk-emulator -storetype pkcs12
-keystore certificat.p12 metronome.apk mobMetronome-app.xml
mobMetronome.swf beep-7.mp3 click.mp3
```

Une invite de commande nous demande de saisir le mot de passe du certificat créé à l'aide de Flash Burrito, puis le traitement se termine en créant l'application. Ainsi, si nous rafraîchissons le répertoire bin nous pouvons voir l'existence d'un nouveau fichier :

metronome.apk



Test du projet

### Fichiers annexes

Ne pas oublier d'inclure les fichiers annexes (mp3, images...) dans les options de compilation

## DÉPLOIEMENT DE L'APPLICATION SUR LE MOBILE VIRTUEL

Copions le fichier .apk précédemment créé dans un nouveau répertoire ayant le chemin suivant : C:\ANDROID\AIR\Metronome\metronome.apk. Pour installer l'application sur le téléphone virtuel nous allons procéder de la même façon que pour l'installation du runtime Adobe Air, via la saisie de cette commande :

```
adb install C:\ANDROID\AIR\Metronome\metronome.apk
```

L'application est à présent installée, nous pouvons procéder à son exécution.

### Exécution de l'application

Voici à présent le moment tant attendu : l'exécution de l'application. Cliquer sur le bouton menu du portable Android, puis sur l'option search. Saisir « mob » et l'application mobMetronome-debug apparaît dans la liste. Cliquer sur l'application pour constater son fonctionnement.

## CONCLUSION

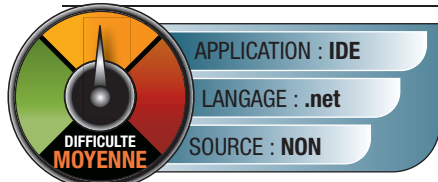
Comme vous l'avez sans doute constaté, la création est assez simple ; mais son déploiement sur un environnement d'émulation demeure encore complexe. Ceci est dû à la jeunesse des produits mais n'ayons crainte, cela devrait s'améliorer très rapidement.

Vous disposez donc à présent de toutes les informations nécessaires à la création d'applications mobiles. Ceci vous permettra de donner de nouvelles perspectives à vos applications Flex.

■ Aurélien Vannieuwenhuyze

# CKlone : l'IDE du Cloud !

La plateforme CKlone est un projet R&D, né de l'ambition de proposer une alternative aux outils de développement plus classiques, pour concevoir et déployer des applications à destination du Cloud. La première brique disponible de cette plateforme en ligne est son Environnement de Développement Intégré basé sur Silverlight, qui permet aux développeurs débutants ou expérimentés de rapidement concevoir et déployer leurs applications pour Windows Azure. Les futures versions sont prévues pour s'intégrer davantage aux outils de développement du marché mais aussi cibler d'autres plateformes de Cloud.



La plateforme CKlone (prononcer « Cyclone ») a été annoncée lors des Microsoft TechDays 2010 par MPoware. L'ambition de ce

projet R&D est de proposer en mode SaaS une plateforme de gestion du cycle de vie des applications pour le Cloud Computing. L'une des premières briques mises en place est l'environnement de développement intégré basé sur Silverlight, une technologie permettant de développer des « Smart Clients » interactifs avec une Interface Homme-Machine riche. L'originalité de cet IDE est d'être accessible directement en ligne depuis n'importe quel navigateur supportant Silverlight (IE, Firefox, Chrome...) et de ne nécessiter aucune installation préalable : L'enregistrement d'un compte utilisateur est cependant nécessaire pour s'authentifier et utiliser le CKlone Smart IDE (ou CKSIDE en abrégé).

## Pourquoi un nouvel outil de développement ?

Le marché des outils de développement est varié : IDE avancés, tels que Visual Studio ou Eclipse, générateurs de code, IDE RAD pour Windows, outils de développement pour SmartPhone, etc. En ce qui concerne le Cloud Computing, les environnements de développement prédominants sont plutôt complexes d'autant que l'ensemble des technologies pour développer une application Cloud est riche et par conséquent compliqué à maîtriser, ce qui rend les projets Cloud coûteux à la vue du résultat.

CKlone apporte une réponse simple et pragmatique en encapsulant la complexité des architectures et applications Cloud dans des composants facilement réutilisables par tout un chacun. La productivité ainsi obtenue permet d'abaisser drastiquement les coûts de développement des applications pour le Cloud, d'en réduire la complexité

et ainsi de permettre aux développeurs CKlone de se concentrer davantage sur l'applicatif que sur la plomberie de cette dernière.

Le modèle de programmation d'une application CKlone repose sur l'architecture du CKlone Application Server qui les exécute. Ce serveur d'applications pour le Cloud est capable de gérer plusieurs instances d'applications sur un même serveur, répartir la charge, gérer l'élasticité, et différents autres aspects caractéristiques des applications du Cloud. Le serveur d'applications sortant du sujet de l'article, nous y reviendrons dans les prochains numéros.

## Comment y accéder ?

Pour accéder au CKlone Smart IDE, rien de plus simple : il suffit de s'enregistrer en ligne sur l'url <http://www.cklone.fr/>. Sur la page d'accueil, cliquer sur le bouton « Register ». Entrez ensuite vos informations d'identification et votre email : ce dernier sera utilisé pour que vous puissiez recevoir le lien d'activation de votre compte. Le mail d'activation peut parfois être remisé dans les courriers indésirables. Une fois votre compte activé, vous pouvez vous authentifier pour accéder au Smart IDE. Etant sur une plateforme bêta, vous aurez noté que le certificat SSL n'est pas validé par votre navigateur car ce certificat est auto-signé. Ce certificat permet de crypter vos données d'authentification : vous pouvez ignorer le message d'alerte de votre navigateur, vos données restent protégées.

Une fois authentifié, le CKlone Smart IDE se charge ! [Fig.1].

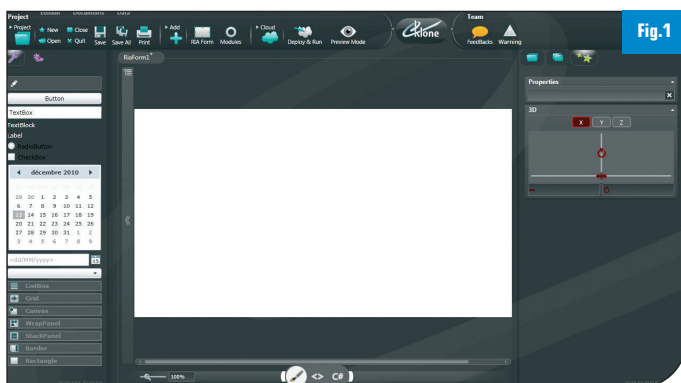
## Les limites de la version Bêta ?

Dans cette première Bêta, l'environnement de développement CKlone permet pas de produire des applications RIA basées sur Silverlight. C# est le langage par défaut mais il est prévu à terme de supporter d'autres langages de la famille .NET mais aussi Java. De même, pour la plateforme de Cloud Computing, bien que Windows Azure soit la seule supportée dans cette première version, d'autres seront proposées à terme telles que Google AppEngine.

## DÉCOUVERTE DU CKLONE SMART IDE

Le Smart IDE est composé de :

- Une zone pour les barres d'outils (chargement/sauvegarde d'un projet, édition, équipe, ...).
- Une toolbox ou boîte à outils de contrôles utilisables dans l'application [Fig.2].
- Un panneau « Effets & Actions » [Fig.3].
- Un panneau à droite pour la gestion des propriétés et de la 3D
- Des panneaux à droite pour la gestion du projet et des éléments de l'application. Par exemple, l'ajout d'images ou de vidéos à l'applica-



tion peut se faire par le Panneau « Application » sous l'onglet Media.

- La zone centrale permet d'accéder au concepteur WYSIWYG, au contenu XAML ou encore au code C#. Les contrôles positionnés sur le concepteur peuvent être redimensionnés, déplacés, supprimés ou encore ajoutés à des contrôles conteneurs, comme le StackPanel ou la Grid par exemple [Fig.4].

## Appliquer un thème à l'application

Les applications CKlone supportent les thèmes : le développeur peut soit utiliser un thème proposé (les thèmes de Microsoft sont fournis par défaut et MPoware fournit aussi des thèmes moins gourmands en ressources), soit charger son propre thème ou un thème élaboré par un designer sous la forme de dictionnaire de ressources XAML. Le choix d'un thème se fait depuis l'éditeur de thème via la boîte à outils et est ensuite appliqué à la boîte à outils : ainsi le développeur voit très exactement l'apparence de son application pendant la phase de conception [Fig.5].

## Exécuter une fonction

Une application CKlone est bâtie sur un modèle qui permet au code d'être exécuté à distance sur le serveur d'applications CKlone, et à l'Interface Homme-Machine d'être affichée en local sur le navigateur de l'utilisateur. Ce type d'application distribuée est très courant sur le Cloud mais peut être complexe à développer. CKlone masque cette complexité via des comportements proposés dans le panneau des « Effets & Actions ». L'une des actions proposées est « **CallMethod** ». Pour l'utiliser, le développeur fait un simple glisser/déplacer de l'action sur l'élément visuel cible, un bouton par exemple [Fig.6]. Lorsque le développeur relâche l'action sur l'élément visuel, un assistant de configuration apparaît : [Fig.7]

Dans le cas de l'action « CallMethod », le développeur peut :

- Sélectionner l'événement à l'origine de l'exécution de la fonction (exemple : le clic sur un bouton),
- Sélectionner une fonction existante ou saisir le nom d'une fonction à créer,
- Ajouter des paramètres à l'appel de fonction avec des valeurs par

défaut ou des expressions de binding (identiques à celles que nous utiliserions avec Expression Blend ou Visual Studio, comme par exemple {Binding Text, ElementName=InputTextBox} qui permet d'envoyer à l'appel de la fonction le texte saisi dans le contrôle InputTextBox).

La validation de l'assistant de cette action génère 2 éléments : le code XAML...

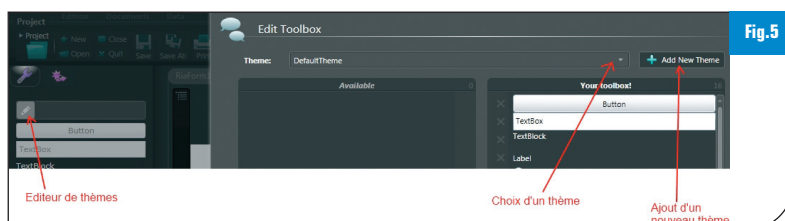
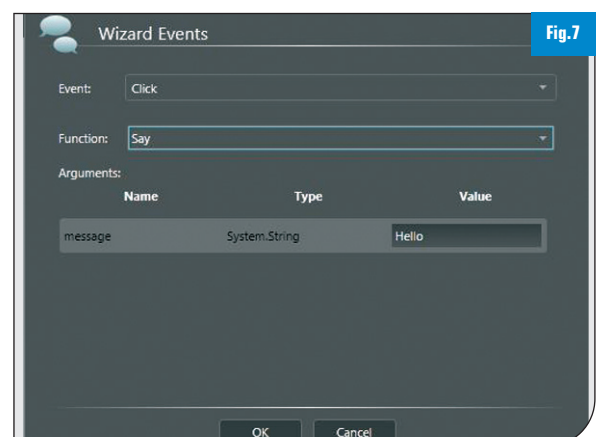
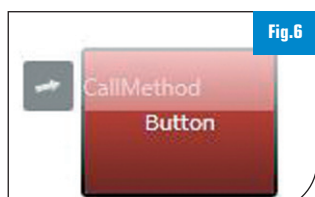
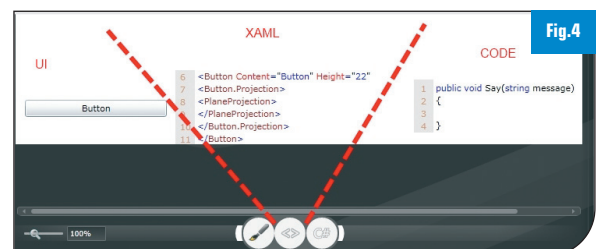
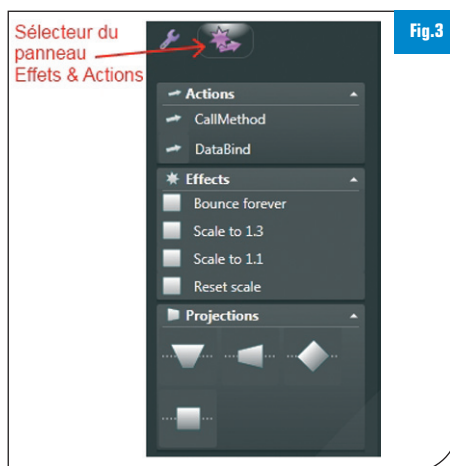
```
<Button Content=»Button« Height=»22« HorizontalAlignment=»Left«
VerticalAlignment=»Top« Width=»182« Canvas.Left=»109« Canvas.
Top=»69«>
  <i:Interaction.Triggers>
    <i:EventTrigger EventName=»Click«>
      <MPowareCKloneRIAActions:Call Method=»Say«>
        <MPowareCKloneRIAActions:Call.Parameters>
          <MPowareCKloneRIAActions:ParameterCollection>
            <MPowareCKloneRIAActions:Parameter Name=»message« Type
=»System.String« Value=»{Binding Text, ElementName=InputTextBox}«>
          </MPowareCKloneRIAActions:Parameter>
        </MPowareCKloneRIAActions:ParameterCollection>
      </MPowareCKloneRIAActions:Call.Parameters>
    </MPowareCKloneRIAActions:Call>
  </i:EventTrigger>
</i:Interaction.Triggers>
</Button>
```

... et le code C# correspondant :

```
public void Say(string message)
{
}
}
```

## Une application multipages

Avec CKlone Smart IDE, le développeur peut ajouter autant de pages (ou RIA Forms) à son application qu'il le souhaite. Pour cela, il utilise l'ajout d'une RIA Form depuis la barre d'outils projet [Fig.8].





Pour renommer une page, le développeur double-clique sur son nom et saisit le nouveau nom d'une page [Fig.9]. Enfin, pour naviguer d'une page à une autre, le développeur peut utiliser une action de type « CallMethod » avec le code suivant :

```
Navigate(«MaPage3»);
```

Remplacer MaPage3 par le nom de votre page.

## Modifier l'IHM depuis le code

Parmi les possibilités des applications développées avec le Smart IDE de CKlone, le développeur peut modifier l'IHM depuis son code exécuté à distance sur le serveur d'applications grâce à la méthode `SetElementProperty`.

Cette méthode prend 3 paramètres :

- Le nom de l'élément visuel à modifier,
- La propriété à affecter,
- Et la valeur à assigner à la propriété.

Par exemple, pour modifier le texte d'un contrôle de type `TextBlock` dont le nom est `OutputTextBlock`, le code sera :

```
SetElementProperty("OutputTextBlock", "Text", "Hello World !");
```

## La liaison aux données

La liaison aux données permet de tirer pleinement profit du `DataBinding` offert par Silverlight mais là aussi d'une manière élégante et simple : Depuis le panneau des effets et actions, il suffit de glisser/déplacer l'action « `DataBind` » sur un contrôle orienté données tel qu'une `ListBox` par exemple. Là encore, l'assistant de configuration de l'action apparaît et permet au développeur de paramétrer la liaison aux données. Ci-dessous, l'exemple d'une liaison à une source de données « `Employees` » possédant 2 colonnes, « `Firstname` » et « `Lastname` » [Fig.10].

Côté code, pour ajouter un enregistrement à une source de données, le développeur pourra utiliser la structure de données « `Record` » avec l'appel à une fonction via l'action « `CallMethod` » :

```
public void AddEmployee(string firstname, string lastname)
{
    Record employee = new Record();
    employee["Firstname"] = firstname;
    employee["Lastname"] = lastname;
    AddRecord("Employees", employee);
}
```

La fonction « `AddRecord` » permet d'ajouter un nouvel enregistrement à la source de données « `Employees` » qui est synchronisée automatiquement avec l'application Silverlight cliente.

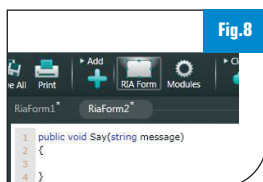


Fig.8

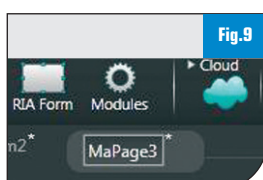


Fig.9

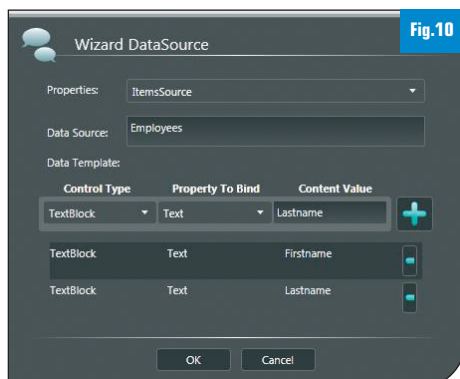


Fig.10

La figure ci-dessous montre une application de gestion des notes de frais complète, basée sur la liaison aux données, avec calcul des différents totaux [Fig.11].

## Ma Première application : Une calculatrice dans les nuages

Dans cette partie, vous apprendrez à créer votre propre calculatrice et à la déployer sur Windows Azure, le tout en quelques minutes !

- Lancer le Smart IDE.
- Sur la partie designer, glisser/déplacer 9 boutons dont vous modifieriez le contenu pour représenter la valeur du bouton (propriété « `Content` » de l'élément « `Button` ») en utilisant le panneau des propriétés.
- Glisser/Déplacer 4 boutons pour les opérations `+`, `-`, `*`, et `/` et un bouton pour calculer le résultat (« `=` »)
- Enfin, glisser/déplacer un élément visuel de type `TextBox` pour afficher les chiffres et résultats des calculs. Modifier sa propriété « `Name` » et entrer « `ResultTextBox` ».

Vous pouvez utiliser le contrôle conteneur que vous souhaitez pour arranger automatiquement les boutons [Fig.12].

Vous êtes libre d'implémenter le code que vous souhaitez pour gérer les traitements de la calculatrice. Le code ci-dessous est fourni à titre d'exemple ; Vous pouvez l'implémenter dans l'éditeur de code.

```
// Fields
private string _number = string.Empty;
private Stack<int> _numbers = new Stack<int>();
private Stack<string> _opers = new Stack<string>();

// Methods
public void Add()
{
    this.PushNumber();
    this._opers.Push("+");
}

public void Divide()
{
    this.PushNumber();
    this._opers.Push("/");
}

public void Equal()
{
    this.PushNumber();
    int num = this._numbers.Pop();
```

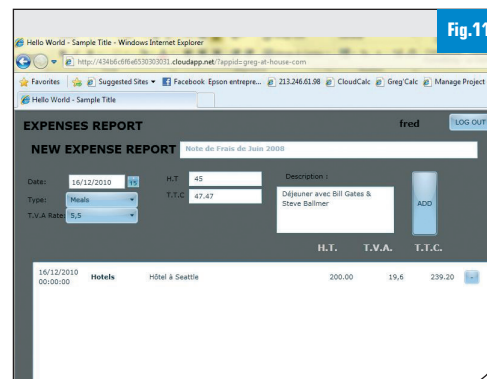


Fig.11



Fig.12

```

int num2 = this._numbers.Pop();
string str = this._opers.Pop();
int num3 = 0;
string str2 = str;
if (str2 != null)
{
    if (!str2 == «/»))
    {
        if (str2 == «*»)
        {
            num3 = num2 * num;
        }
        else if (str2 == «-»)
        {
            num3 = num2 - num;
        }
        else if (str2 == «+»)
        {
            num3 = num2 + num;
        }
    }
    else
    {
        num3 = num2 / num;
    }
}
base.SetElementProperty(«ResultTextBox», «Text», num3.ToString());
this._number = num3.ToString();
}

public void Multiply()
{
    this.PushNumber();
    this._opers.Push(«*»);
}

public void Push(string number)
{
    this._number = this._number + number;
    base.SetElementProperty(«ResultTextBox», «Text», this._number);
}

private void PushNumber()
{
    this._numbers.Push(int.Parse(this._number));
    this._number = string.Empty;
}

```

```

public void Subtract()
{
    this.PushNumber();
    this._opers.Push(«-»);
}

```

Il ne reste plus qu'à utiliser les actions de type « CallMethod » pour relier les boutons aux fonctions appropriées. Pour cela, glisser/déplacer l'action « CallMethod » depuis le panneau « Effets & Actions » autant de fois qu'il est nécessaire pour relier tous les boutons aux fonctions. Pour les chiffres, saisir en paramètre les chiffres correspondants aux nombres, comme indiqué sur la figure [Fig.13]. Pour les fonctions de calcul, sélectionner la fonction appropriée. Sauvegarder le projet ("Save All" depuis la barre d'outils Projet) et cliquer le bouton « Build & Deploy ». Cette action compile le code du projet, déploie les ressources applicatives sur un espace de stockage Windows Azure et exécute le client Web CKlone pour Windows Azure. Le client Web CKlone pour Windows Azure est un runtime client permettant de charger et d'exécuter les applications CKlone connectées au serveur d'applications CKlone.

Vous pouvez maintenant modifier et enrichir votre calculatrice à volonté ou simplement changer son apparence en quelques clics !

## CONCLUSION

Les premières versions de CKlone avec son Smart IDE permettent de développer et déployer très rapidement des applications RIA à destination du Cloud et notamment de la plateforme Microsoft Windows Azure. Les développeurs pragmatiques, mais aussi les experts, trouveront dans cet IDE nouvelle génération, accessible en ligne, un accélérateur de développement d'applications pour le Cloud. De nouvelles versions du CKlone Smart IDE sont mises en production toutes les 1 à 2 semaines. Chaque mise en production est suivie d'une annonce dans le groupe d'utilisateurs CKlone sur Facebook. Toute l'équipe R&D travaille à l'amélioration constante du Smart IDE et de la plateforme CKlone. Une dizaine d'applications représentatives, telles qu'une application de notes de frais, un jeu de Sudoku, un lecteur de flux RSS, un site de commerce électronique, etc., sont utilisées pour produire des exemples de code, des web cast et tutoriels.

Vous aussi, devenez membre du groupe d'utilisateurs CKlone et participez en donnant votre feedback sur des améliorations possibles. Vous aurez ainsi accès aux dernières annonces et ressources concernant la plateforme.

## Références

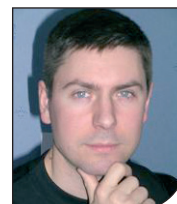
Le site de CKlone : <http://www.cklone.fr/>

Le groupe utilisateurs CKlone sur Facebook :

<http://www.facebook.com/group.php?gid=114390445252061>

Vidéo d'introduction de CKlone :

[http://www.youtube.com/watch?v=6VG\\_vwHH7HA](http://www.youtube.com/watch?v=6VG_vwHH7HA)



■ Frédéric Queudret

MVP Client Application & CTO – MPoware.

MPoware est une société française d'édition de logiciels et de prestations de services sur les technologies .NET.

<http://www.mpoware.com/> - [contact@mpoware.com](mailto:contact@mpoware.com)

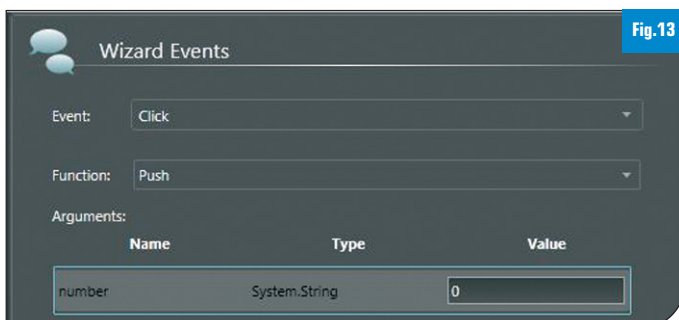
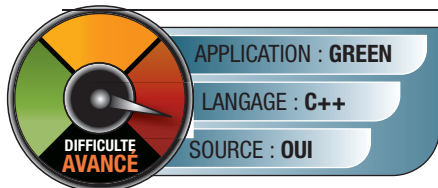


Fig.13

# Ajuster la fréquence de votre processeur avec l'API Windows Power Management

Méconnue, l'API Power Management de Windows permet une gestion très fine de la consommation d'énergie d'un PC. Nous la mettons en œuvre aujourd'hui pour ajuster la cadence du processeur en fonction des besoins.



Sans doute avez-vous déjà visité les options d'alimentation dans le panneau de configuration de Windows ? Rien de très folichon là-dedans pour les ordinateurs

de bureau. C'est un peu plus intéressant pour les ordinateurs portables qui disposent de deux modes d'alimentation, secteur et batterie. Ce panneau de configuration est toutefois bien loin de laisser entrevoir la richesse et la puissance de l'API qui se cache derrière. Avec cette API, il est possible d'ajuster de très nombreux paramètres inédits. Pour le découvrir, nous imaginerons que nous possédons un ordinateur portable que nous devons utiliser pendant un temps très long, mais dans une situation demandant très peu de puissance, comme par exemple, la simple prise de notes au cours d'une conférence interminable. Il serait intéressant d'avoir dans cette situation une petite application de type tableau de bord, qui informe de l'autonomie de la batterie, ce qui reste classique, mais qui permette aussi d'ajuster la fréquence du processeur, ou de fixer à cette fréquence un seuil maximum, afin d'économiser la batterie. Ce dernier point est beaucoup plus geek, et cet article va s'efforcer de donner tous les éléments nécessaires à la réalisation d'une telle application.

## 1 LE MATÉRIEL

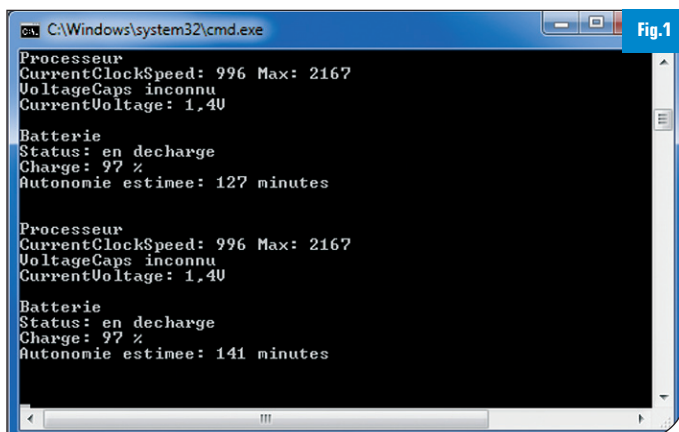
Il va de soi qu'il est nécessaire de posséder une machine récente qui supporte l'underclocking, ou sous-cadencement du processeur par programmation. Les exemples qui accompagnent cet article et que vous trouverez complets sur notre site ont été essayés sur un ordinateur portable ASUS X71SL. Le code C++ a été écrit sous Visual Stu-

dio 2008. Tout autre compilateur C++ conviendra. Le système d'exploitation est Windows 7. Les exemples devraient fonctionner également avec Windows Vista. Ils ne fonctionneront pas sur des systèmes antérieurs. Enfin, signalons que l'API Power Management est constituée de deux jeux de fonctions pêle-mêle, pré Vista et post Vista. Le lecteur qui voudrait aller plus loin et enrichir les exemples de cet article devra être très attentif à n'utiliser que des fonctions Vista ou supérieures. Sauf erreur de ma part, les fonctions pour les systèmes antérieurs ne fonctionnent pas très bien sous Vista et Windows 7. Enfin on ne peut se passer de la documentation Microsoft, MSDN en ligne ou SDK 7, même si cette documentation manque cruellement de clarté, ce à quoi cet article va tenter de remédier.

## 2 SAVOIR OBSERVER LE SYSTÈME

Si l'on souhaite ajuster la consommation énergétique d'un système, il faut commencer par être capable de savoir à tout moment ce qui s'y passe. Sous Windows, il y a une API pour cela, la WMI, qui est l'implémentation des spécifications CIM et WBEM définies par la Distributed Management Task Force. Celle-ci est une organisation qui développe et maintient des standards pour l'administration de systèmes informatiques. Dit très succinctement, CIM est un jeu de classes, chacune d'entre elles décrivant un élément d'un système, et la WBEM est un jeu de fonctions pour manipuler ces classes. Le sujet de la WMI a été traité par exemple dans l'article «Observez la charge de votre processeur multi-cœur sous Windows» de Programmez! n° 110 et nous invitons le lecteur qui voudrait approfondir la question à s'y reporter. Nous donnons simplement ici un programme en C# (CSharpWatchProcessor disponible sur notre site) qui affiche la cadence de processeur, la nature de l'alimentation (secteur ou batterie) et l'état de charge de la batterie ainsi que son autonomie estimée. Il est nécessaire que le programme fonctionne quelques secondes pour que la valeur de l'autonomie soit ajustée par le système : [Fig.1].

Quelques petites remarques à propos de ce code. Tout d'abord ne nous laissons pas abuser par la terminologie : des termes comme ManagementClass concernent la WMI et sont sans rapport avec l'API Power Management qui est au centre de cet article. Sur ma machine, il s'est avéré impossible de lire la propriété VoltageCaps de la classe WIN32\_Processor. La lecture provoque la levée d'une exception, et c'est pourquoi le morceau de code correspondant est placé dans une construction try/catch. Bien entendu, il est possible d'interroger la WMI avec C++, mais cela fait appel à des notions qui sortent du cadre de cet article, nous renvoyons pour cela le lecteur à Programmez! 110. Le code en C# est également plus concis. Pourquoi avoir mentionné WMI dans un article dédié à l'API Power Management ? Parce que WMI présente un énorme avantage: elle



Monitoring du cadencage du microprocesseur et de l'état de la batterie à travers la WMI.

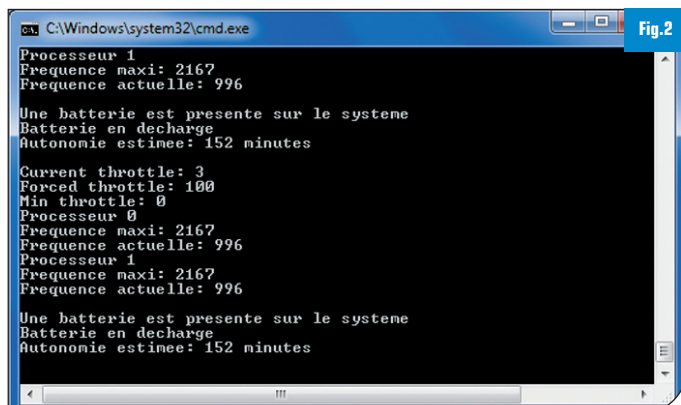


est unifiée. Quand on sait l'utiliser pour surveiller un élément du système, on sait faire de même pour n'importe quel élément. En outre, les données observées avec elle permettent de vérifier qu'un programme écrit avec d'autres API se comporte correctement. Nous abordons maintenant l'API Power Management.

### 3 AVEC L'API POWER MANAGEMENT

Tout programme C ou C++ qui travaille avec cette API doit voir son édition de liens faite avec `PowrProf.lib`. Pour l'état du système, nous utilisons cette fois une de ces fonctions «couteau suisse» dont Microsoft a le secret. Il s'agit de `CallNtPowerInformation` qui, comme le nom le suggère, est antérieure à Windows Vista. Cette API renvoie des résultats qui ne sont pas toujours très fiables. Ainsi, si pour nous faire la main, nous lui demandons si la cadence de notre processeur est ajustable (Programme `DemoCallNtPowerInformation` sur notre site). La fonction nous répond que non... et du coup nous ne pouvons obtenir les seuils d'ajustement mini et maxi que cette fonction devrait normalement nous fournir. Fort heureusement, lorsqu'il s'agit de lire la cadence du processeur ou l'état de la batterie, les choses se passent beaucoup mieux. [Fig.2] (`CppWatch-Processor` sur notre site) :

Dans ce code, on notera la présence de la définition de la structure `PROCESSOR_POWER_INFORMATION`. Normalement celle-ci devrait se situer dans les fichiers en-tête du SDK. Mais la documentation Microsoft nous avertit que cette structure a été oubliée, d'où l'ajout dans le code. Bien sûr, le jour où Microsoft aura corrigé ses fichiers en-tête, le code ne compilera plus :) Le lecteur qui a lu l'article «Observez la charge de votre processeur multi-cœur sous Windows» cité plus haut sait que, dans le cas d'une machine équipée d'un processeur multi-cœur, la WMI ne voit qu'un processeur et des cœurs, ce qui est logique. Avec `CallNtPowerInformation`, les choses fonctionnent à l'inverse. C'est pourquoi notre code questionne la fonction `Win32 GetSystemInfo` pour connaître le nombre de «processeurs». On utilise ensuite ce résultat pour allouer un tableau de `PROCESSOR_POWER_INFORMATION`. Le reste du code est trivial. Comme on le voit dans l'exemple ci-dessous `CallNtPowerInformation` peut être utilisée pour connaître l'état de la batterie. Cependant, si l'on veut s'amuser à écrire du code aussi «geek» que possible, on notera l'alternative qui consiste à interroger le driver de la batterie avec la fonction `DeviceloControl`. Le lecteur trouvera sur notre site l'exemple `DemoBatterieIoControl` qui peut servir de point de départ à l'écriture d'un code adapté aux besoins. Cet exemple est directe-



```

C:\Windows\system32\cmd.exe
Processeur 1
Frequence maxi: 2167
Frequence actuelle: 996

Une batterie est presente sur le systeme
Batterie en decharge
Autonomie estimee: 152 minutes

Current throttle: 3
Forced throttle: 100
Min throttle: 0
Processeur 0
Frequence maxi: 2167
Frequence actuelle: 996
Processeur 1
Frequence maxi: 2167
Frequence actuelle: 996

Une batterie est presente sur le systeme
Batterie en decharge
Autonomie estimee: 152 minutes

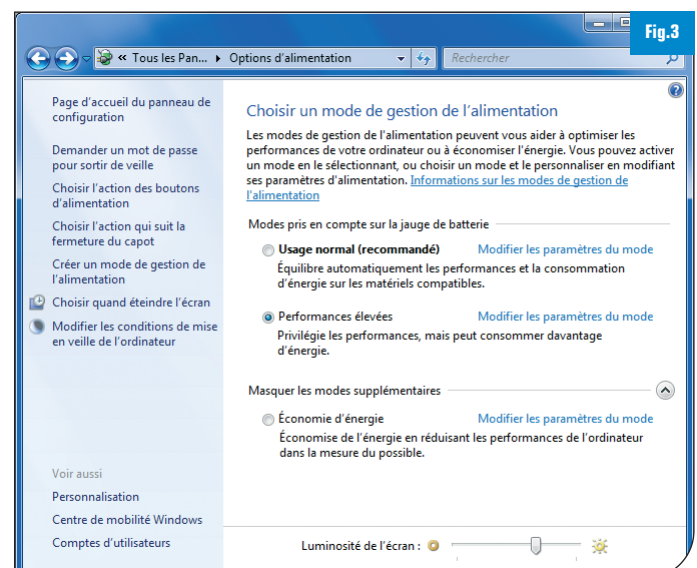
```

Monitoring du cadencage du microprocesseur et de l'état de la batterie à travers l'API Power Management.

ment tiré du SDK Windows et n'est pas reproduit ici. Enfin, en ce qui concerne l'alimentation du PC, on pourrait encore interroger la fonction `GetSystemPowerStatus`. Voir `DemoGetSystemPowerStatus` sur notre site.

### 4 LES PLANS DE CONSOMMATION ÉNERGÉTIQUE

Depuis Windows Vista les paramètres relatifs à la consommation d'énergie du PC sont regroupés dans un ensemble de données appelé plan (ou Schemes dans la documentation). Par défaut, Windows propose 3 plans [Fig.3]. Un mode (ou plan) pour de bonnes performances, un mode pour de bonnes économies, et un mode équilibré. Il est possible d'ajouter de nouveaux plans, éventuellement de les supprimer ultérieurement depuis l'interface du panneau. Et bien sûr, cela est également possible par programmation, via l'API Power Management. Ce panneau ne laisse pas entrevoir toute la richesse des réglages, même lorsque l'on clique sur les options avancées d'un plan. Ainsi l'illustration [Fig.4] ne montre que 3 paramètres relatifs au processeur. Nous allons bientôt découvrir qu'il en existe beaucoup plus. Les plans de gestion d'énergie ont une structure hiérarchique. Au sommet, le plan lui-même. Ce plan contient des sous-groupes de données gérant chacun un composant de l'ordinateur. Il existe par exemple, dans un plan, un sous-groupe pour le processeur, un sous-groupe pour l'écran, un sous-groupe pour les disques, un sous-groupe pour la batterie, etc. Chaque sous-groupe contient un ensemble de valeurs de réglages, appelées settings dans la documentation. Les plans, que ce soit ceux par défaut ou qu'il s'agisse de plans ajoutés ultérieurement, sont identifiés par un GUID, ou Globally Unique Identifier. Les sous-groupes sont à leur tour identifiés par des GUID. Enfin les valeurs sont, elles, marquées par des GUID. Il est important de bien garder à l'esprit que si chaque plan est associé à un GUID, par définition unique, les GUID des sous-groupes sont les mêmes dans tous plans. Il en va de même pour les settings. Enfin, les GUID des plans par défaut sont donnés dans la documentation, ainsi que les GUID des sous-groupes. Sauf erreur de votre serveur, les GUID des settings ne sont pas documentés, mais on les obtient facilement en écrivant un peu de code. Un GUID est un type de données de 16 octets, ainsi défini (en-tête `guiddef.h`) :



Par défaut Windows propose 3 plans de gestion de l'énergie.

```
typedef struct _GUID {
    unsigned long   Data1;
    unsigned short  Data2;
    unsigned short  Data3;
    unsigned char   Data4[ 8 ];
} GUID;
```

Mais on le connaît plus souvent sous cette forme :

```
{370F4BE5-D0FE-4128-9854-DE54765376CA}
```

Une macro, `DEFINE_GUID`, déclarée dans l'en-tête `initguid.h` permet de convertir cette forme en la structure donnée plus haut.

## 5 ENUMÉRER LES PLANS

Voici un programme (DemoPowerEnumeratesur notre site) qui énumère tous les plans d'énergie présents sur une machine. Ainsi qu'on le voit sur l'illustration ci-dessous [Fig.5], on retrouve les GUID par défaut donnés par la documentation. En outre il est possible d'obtenir les chaînes de textes descriptives associées. Ce moyen est a priori le seul pour savoir à quoi servent les settings non documentés.

```
#include <iostream>
using namespace std;
#include <windows.h>
#include <PowrProf.h>

#define BIGSIZE 1024
DWORD bigsize = BIGSIZE;

void ConvertString(WCHAR* src, CHAR* dest) {
    BOOL UsedDefault;
    WideCharToMultiByte(CP_OEMCP,
        WC_NO_BEST_FIT_CHARS,
        src, -1, dest, BIGSIZE, « », &UsedDefault);
}

void PrintGuid(GUID* guid) {
    cout << hex;
    cout << guid->Data1 << "-" << guid->Data2 << "-" << guid->Data3;
```

```
for(int i=0;i<8;i++)
    cout << (int)guid->Data4[i];
cout << endl;
}

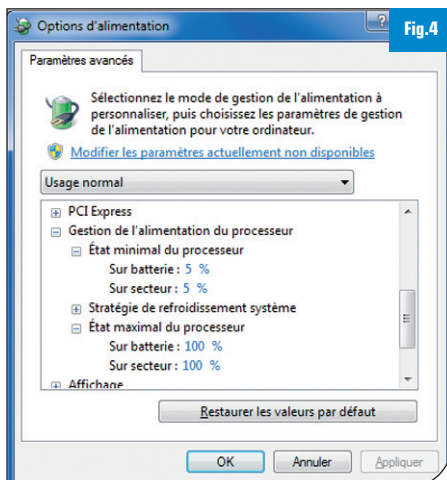
void PrintPowerScheme(GUID* guid) {
    DWORD result;
    UCHAR buffer[BIGSIZE];
    CHAR chBuffer[BIGSIZE];

    result = PowerReadDescription(NULL,
        guid, NULL, NULL, buffer, &bigsize);

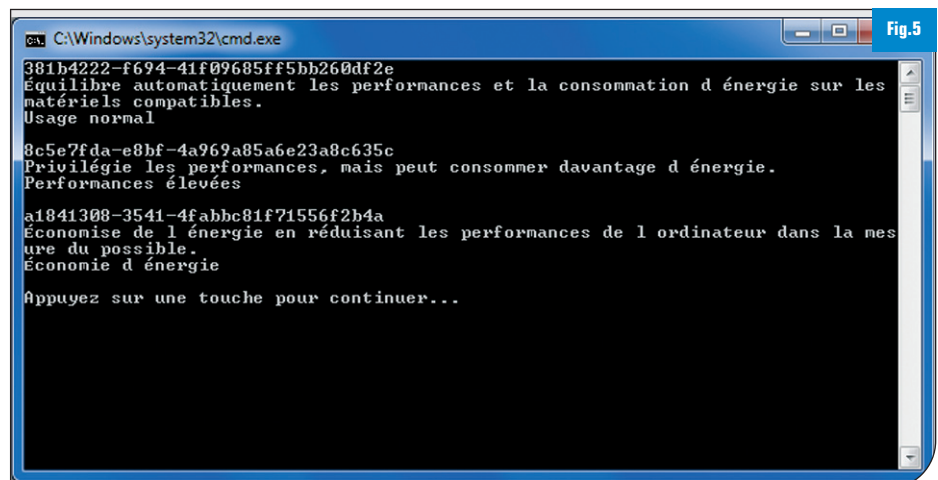
    if(!result) {
        ConvertString((WCHAR*)buffer, chBuffer);
        cout << chBuffer << endl;
    }
    bigsize = BIGSIZE;
    result = PowerReadFriendlyName(NULL,
        guid, NULL, NULL, buffer, &bigsize);

    if(!result) {
        ConvertString((WCHAR*)buffer, chBuffer);
        cout << chBuffer << endl;
    }
    bigsize = BIGSIZE;
}

int main(int argc, char* argv[]) {
    UCHAR* buffer;
    DWORD buffersize;
    ULONG index;
    DWORD result;
    index = 0;
    result = 0;
    while(1) {
        result = PowerEnumerate(NULL,
            NULL, NULL, ACCESS_SCHEME, index,
            NULL, &buffersize);
```



Les options avancées d'un plan ne reflètent pas la richesse des réglages possibles par programmation.



Énumération des plans de gestion de l'énergie.

```

if(result != 0 && result != ERROR_MORE_DATA)
    break;

buffer = (UCHAR *)LocalAlloc(LPTR, buffersize);
result = PowerEnumerate(NULL,
    NULL, NULL, ACCESS_SCHEME, index, buffer, &buffersize);
if(result) {
    LocalFree(buffer);
    break;
}

PrintGuid((GUID*)buffer);
PrintPowerScheme((GUID*)buffer);
cout << endl;
LocalFree(buffer);
index++;
}
return 0;
}

```

La plupart des fonctions «Power» doivent être invoquées deux fois consécutivement. Une première fois pour obtenir la taille du buffer qui sera rempli par le second appel. La première partie du programme (dans main) alloue la mémoire à chaque fois. Ensuite par souci de simplicité, de concision et aussi par paresse, nous utilisons de gros tampons déclarés sur la pile.

## 6 CRÉER UN NOUVEAU PLAN

Pour expérimenter de nouveaux réglages, il est pertinent de créer un nouveau plan. Faire cette opération entièrement et à partir de zéro n'est pas envisageable, nous ne connaissons même pas tous les settings existants. La technique consiste donc à dupliquer un des plans par défaut au moyen de la fonction `PowerDuplicateScheme`. Cette fonction génère ou reçoit le GUID du plan à créer. Le plus pratique est d'utiliser l'outil `guidgen.exe` ou `uuidgen.exe` de la SDK Windows pour obtenir un GUID humainement lisible, puis de l'utiliser comme illustré ci-dessous. Le code est extrait de `DemoPowerDuplicateScheme`, disponible sur notre site.

```

#include <initguid.h> // pour le DEFINE_GUID ci-dessous

// {370F4BE5-D0FE-4128-9854-DE54765376CA}
DEFINE_GUID(PROGRAMMEZGUID,
0x370f4be5, 0xd0fe, 0x4128, 0x98, 0x54, 0xde, 0x54, 0x76, 0x53,
0x76, 0xca);

// etc.

GUID leguid = PROGRAMMEZGUID;
GUID sous_guid = NO_SUBGROUP_GUID;
GUID* active;
GUID* target = &leguid;

cout << «Plan actif» << endl;
PowerGetActiveScheme(NULL, &active);
PrintGuid(active); // voir programme complet
cout << endl;

```

```

result = PowerDuplicateScheme(NULL,
    active,
    &target);

if(result == ERROR_ALREADY_EXISTS)
    cout << «Le plan existe déjà» << endl;
if(!result)
{
    WCHAR texte[] = L«Un bon plan Programmez ! Abonnez vous ! :-»»;
    PowerWriteDescription(NULL,
        target,
        NULL,
        NULL,
        (UCHAR*)texte,
        sizeof(texte)+2);
    cout << «succes duplication» << endl;
}
cout << endl;

```

## 7 OBTENIR UNE DESCRIPTION DES SETTINGS

Tous les settings d'un sous-groupe d'un plan peuvent être énumérés. On passe par l'API `PowerEnumerate`, mais cette fois en lui imposant un GUID de plan et un GUID de sous-groupe. La fonction énumère alors les settings, qu'il suffit d'afficher. Le code ci-dessous, est extrait de notre exemple `DemoReadProcessorSettings` qui énumère toutes les settings relatifs au processeur. Ce code affiche les chaînes descriptives et la valeur du setting pour une alimentation secteur. On emploie pour cela la fonction `PowerReadACValue`. Si on voulait la valeur pour une alimentation avec la batterie, on emploierait `PowerReadDCValue` tout simplement.

```

void PrintPowerSetting(GUID* leguid, GUID* sous_guid, GUID* setting) {
    DWORD result;
    UCHAR buffer[BIGSIZE];
    CHAR chBuffer[BIGSIZE];
    ULONG letype;

    result = PowerReadDescription(NULL,
        leguid, sous_guid, setting, buffer, &bigsize);

    if(!result) {
        ConvertString((WCHAR*)buffer, chBuffer);
        cout << chBuffer << endl;
    }
    bigsize = BIGSIZE;

    result = PowerReadFriendlyName(NULL, leguid,
        sous_guid, setting, buffer, &bigsize);

    if(!result) {
        ConvertString((WCHAR*)buffer, chBuffer);
        cout << chBuffer << endl;
    }
    bigsize = BIGSIZE;

    result = PowerReadACValue(NULL, leguid, sous_guid,

```



```

    setting, &letype, buffer, &bigsize);

cout << dec;
switch(letype) {
case REG_DWORD:
    cout << «Valeur entiere: » << (int)(*buffer) << endl;
    break;
case REG_SZ:
    cout << «Valeur chaine: » << buffer << endl;
    break;
default:
    cout << «Type non reconnu» << endl;
    break;
}
bigsize = BIGSIZE;
}

```

Et surprise ! comme on le voit sur l'illustration, les settings du processeur sont extrêmement nombreux [Fig.6]. Dans tout ce fatras, dont la découverte est, il faut le dire, passionnante, il suffit, pour poursuivre notre expérimentation, de repérer le GUID du setting définissant la limite supérieure de la cadence du processeur. Ce GUID est :

```
{bc5038f7-23e0-496096da33abaf5935ec}
```

## 8 MODIFIER LA VALEUR D'UN SETTING

Nous atteignons enfin notre but. Voici le code qui modifie, pour l'alimentation secteur, la valeur maximum de la cadence du processeur dans le plan que nous venons de créer, et qui définit ce plan, comme le plan actif. La valeur doit être exprimée en pourcentage :

```

// DemoPowerWriteACValue.cpp
#include <windows.h>
#include <PowrProf.h>
#include <initguid.h> // pour les DEFINE_GUID ci-dessous

// {370F4BE5-D0FE-4128-9854-DE54765376CA}
DEFINE_GUID(PROGRAMMEZGUID,
0x370f4be5, 0xd0fe, 0x4128, 0x98, 0x54, 0xde, 0x54, 0x76, 0x53,
0x76, 0xca);

// {bc5038f7-23e0-496096da33abaf5935ec}
DEFINE_GUID (SETTINGPROCGUID,
0xbc5038f7, 0x23e0, 0x4960, 0x96, 0xda, 0x33, 0xab, 0xaf, 0x59,
0x35, 0xec);

#define CADENCEMAX 70 // 70% maxi
int main(int argc, char* argv[]) {
    GUID leguid = PROGRAMMEZGUID;
    GUID sous_guid = GUID_PROCESSOR_SETTINGS_SUBGROUP;
    GUID setting = SETTINGPROCGUID;
    DWORD index;

    PowerReadACValueIndex(NULL,
        &leguid, &sous_guid, &setting, &index);

```

```

    PowerWriteACValueIndex(NULL,
        &leguid, &sous_guid, &setting, CADENCEMAX);

    PowerSetActiveScheme(NULL, &leguid);
    return 0;
}

```

L'expérimentation montre que la fréquence d'horloge ne suit pas strictement la valeur donnée, mais il est toutefois possible de réduire la cadence de plus de la moitié. Voici les résultats obtenus par votre serveur sur sa machine :

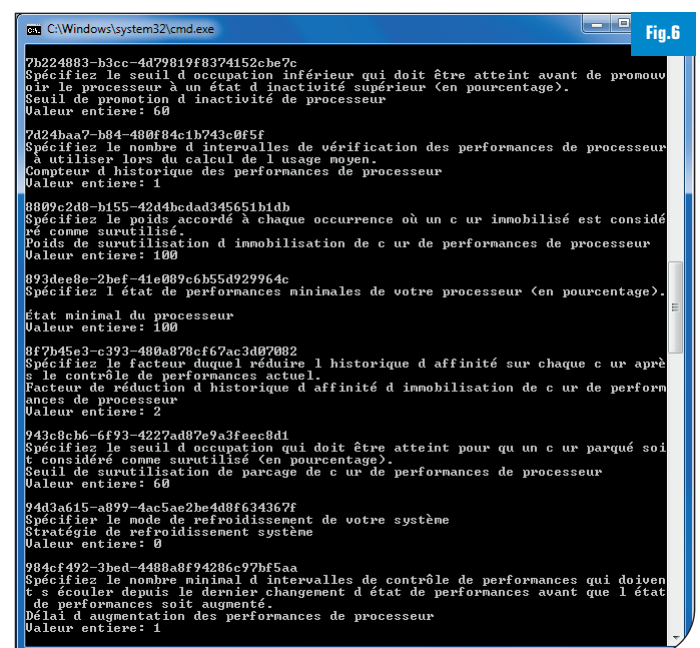
Pourcentage	Fréquence réelle
100 %	2167 MHz
80% - 90% - 95%	1646 MHz
65% - 70% - 75%	1321 MHz
20% - 40% - 60%	996 MHz

Ces résultats montrent également que le système refuse que l'on descende en dessous d'un certain seuil. C'est normal, car un processeur trop underclocké risque de devenir instable.

## 9 CONCLUSION

A partir des éléments donnés dans cet article, il est possible de construire une application 'geek' en diable. Vous pouvez par exemple choisir la stratégie de refroidissement du système ou le taux d'inactivité nécessaire pour que le processeur ajuste sa cadence entre deux limites mini et maxi. Enfin, outre le processeur, vous pouvez adapter les morceaux de code donnés pour toucher à tout: intensité de l'éclairage de l'écran, le temps d'inactivité d'un disque avant son arrêt, etc. :-)

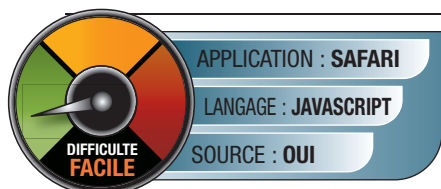
■ Frédéric Mazué - [fmazue@programmez.com](mailto:fmazue@programmez.com)



Les settings de gestion du processeur sont extrêmement nombreux.

# Ecrire des extensions pour les navigateurs Opera et Safari

Après avoir examiné le mois dernier comment écrire une extension pour le navigateur Google Chrome, nous voyons aujourd'hui comme faire le même travail pour les navigateurs Opera et Safari.



Nous le savons bien, les éditeurs de navigateurs se livrent une guerre sans merci. Tous essaient d'améliorer la prise en charge des standards du Web, ce qui est une excellente chose, tous sont concernés par l'amélioration de la sécurité et des performances, ce qui est également très bien. Mais l'expérience a montré que cela ne suffisait pas à rendre un navigateur populaire. Un navigateur ne peut sortir du lot que lorsqu'il propose un mécanisme d'extensions.

Ainsi Google Chrome n'a vraiment décollé qu'avec la version 5 qui proposait cette fonctionnalité dont Internet Explorer et Firefox disposent depuis longtemps. Après avoir étudié les extensions sous Google Chrome 5 le mois dernier, nous examinons aujourd'hui comment les choses se passent avec les deux autres outsiders : Opera et Safari. Les exemples accompagnant cet article ont été testés avec les versions Windows de ces navigateurs. Les sources sont disponibles sur notre site.

## 1 OPERA

Ce navigateur développé en Norvège qui se distingue par un moteur Javascript extrêmement rapide, présente depuis la version 8, un mécanisme d'extension que l'on peut qualifier d'assez rudimentaire par rapport à la concurrence. A partir de la version 11, encore en bêta au moment de la rédaction de cet article, Opera disposera en outre d'un mécanisme d'extensions à base de HTML, CSS et Javascript, similaire à ce que proposent ses concurrents.

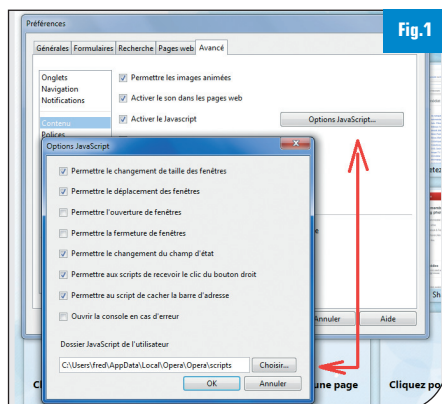
En attendant c'est dans la simplicité du mécanisme actuel, qui existait toujours dans la version 11, que se trouve sa plus grande originalité. En effet une extension Opera n'est ni plus ni moins qu'un script Javascript, dit script utilisateur, chargé avant la page, et qui

s'exécute dans la portée globale. Concrètement, cela signifie qu'un script incorporé normalement à une page et une extension partageant le même espace et donc partagent les variables, les fonctions, etc. Ceci implique qu'une page maligne peut voler des informations sensibles que le développeur de l'extension aurait pu laisser traîner dans son code. Autre originalité, il existe un mécanisme d'événement qui permet d'anticiper l'exécution d'un script de la page chargée. Le gestionnaire de cet événement peut alors accéder au code et le modifier totalement: ajouter des variables, des fonctions, les modifier, ou même réécrire le code en totalité ou encore le passer à la trappe. Opera est encore capable d'exécuter la plupart des scripts Greasemonkey.

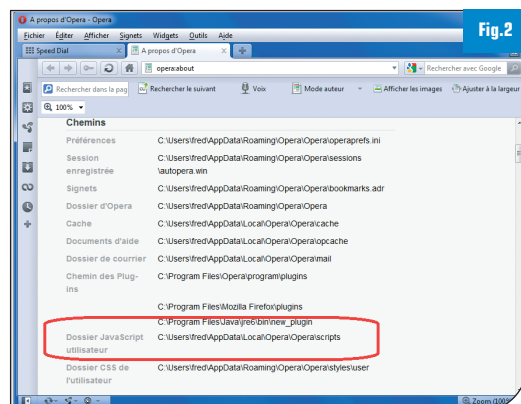
## 2 HELLO WORLD!

Par défaut, sous Opera, le mécanisme d'extensions est inactif. Pour l'activer il faut et il suffit de définir un répertoire dans lequel seront déposés les scripts utilisateur. Les extensions peuvent alors être installées à chaud, ce qui signifie qu'il n'est pas nécessaire de relancer Opera pour qu'elles soient prises en compte.

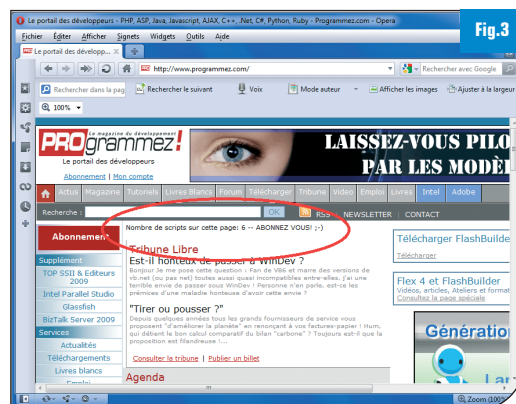
Pour définir le répertoire, il faut sélectionner 'Préférences' dans le menu du navigateur, puis sous l'onglet 'Avancé/Contenu', cliquer sur le bouton 'Options Javascript'. Il est alors possible de définir le répertoire pour les scripts, comme le montre l'illustration ci-dessous [Fig.1]. Si tout est en ordre, le répertoire ainsi défini sera listé dans l'onglet 'A propos d'Opera', accessible depuis le menu, et comme illustré ci-dessous [Fig.2]. Pour notre Hello World, nous voulons simplement que notre extension ouvre une alerte lorsque l'utilisateur navigue sur la page d'accueil du site de Programmez!. Pour toute autre page du site ou pour tous les autres sites, l'extension restera silencieuse. Pour filtrer ainsi les pages sur lesquelles agir ou non, on place en tout début du script des directives include ou exclude. Voici ce que cela donne pour notre exemple :



Définition du dépôt de scripts utilisateur sous Opera



Notre dépôt apparaît maintenant dans l'onglet 'A propos d'Opera'.



Notre script Opera en action sur la page d'accueil de Programmez!

```
// ==UserScript==
// @include http://www.programmez.com/
// @exclude http://www.programme.com/*
// ==/UserScript==

alert("Hello World Programmez!");
```

Si aucune directive n'est placée dans l'en-tête, le script s'exécute sur toutes les pages rencontrées. Comme nous l'avons mentionné plus haut, le fait qu'un script d'extension s'exécute dans la portée globale, conjointement aux scripts de la page peut poser un problème de sécurité.

Si l'on veut éviter qu'un script d'une page ne "vole" des informations à une extension, on englobera tout le code de celle-ci dans une fonction anonyme, comme ceci :

```
// ==UserScript==
// @include http://www.programmez.com/
// @exclude http://www.programme.com/*
// ==/UserScript==

(function() {
    alert("Hello World Programmez!");
})();
```

### 3 AGIR SUR LE DOM

Comme avec tous les navigateurs, les extensions Opera peuvent agir sur le DOM du document. Avec Opera on peut même ne faire que cela, mais c'est à la fois remarquablement facile et puissant. Voici un script qui compte le nombre de scripts Javascript dans la page d'accueil de Programmez! puis qui modifie le div contenant les actualités dans la page, comme illustré [Fig.3].

```
// ==UserScript==
// @include http://www.programmez.com/
// @exclude http://www.programme.com/*
// ==/UserScript==

(function() {

var nb_script = 0 ;

window.opera.addEventListener('BeforeScript', function (e)
    {
        nb_script++;
    }, false);

function showResult()
{
    res = "Nombre de scripts sur cette page: ";
    res += nb_script;
    res += " -- ABONNEZ VOUS! ;-)";
    div_elem = document.getElementById('actus_accueil');
    div_elem.innerHTML = res;
}

})();
```

```
window.onload = showResult;

})();
```

Tout est emballé dans une fonction anonyme, comme expliqué précédemment. Nous commençons par définir un gestionnaire pour l'événement 'BeforeScript' qui est déclenché à chaque fois que le moteur Javascript s'apprête à exécuter un script dans la page. C'est cela qui nous permet de compter le nombre de scripts. Nous sommes sûrs de tous les compter car il est garanti que le script d'extension est chargé avant la page visitée. En revanche et pour la même raison, lorsqu'il s'agit de modifier le DOM de la page, nous devons nous assurer que celle-ci est bien complètement chargée. Pour cela nous utilisons le gestionnaire classique window.onload. L'essentiel est dit en ce qui concerne les extensions d'Opera, mais nous n'avons pas épuisé toutes les possibilités. Par exemple, dans le gestionnaire de l'événement 'BeforeScript', nous aurions pu modifier le code des scripts au moyen des API. Nous renvoyons pour cela le lecteur à la documentation.

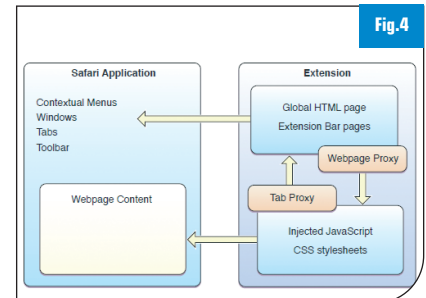


Fig.4 Organisation d'une extension Safari. Ce diagramme est issu de la documentation officielle.

### 4 SAFARI

Si avec Opera le mécanisme d'extensions est rudimentaire et se limite à une injection de scripts susceptible de faire le bonheur du bidouilleur, avec Safari, nous trouvons une architecture et une API complexes et élaborées, plus proches de ce que nous avons vu le mois dernier avec le navigateur Google Chrome. Comme pour Chrome, on travaille avec HTML, CSS et Javascript, mais les possibilités sont plus riches. En effet, il est possible de définir des contrôles par l'ajout de boutons dans la barre d'outil, ajouter des entrées dans le menu contextuel, créer des barres d'extensions, en plus de la classique injection de script.

Une extension Safari est constituée de 2 parties. Une partie interagit avec le navigateur lui-même, l'autre avec la page Web visitée. Les deux parties sont strictement cloisonnées et ne peuvent communiquer entre elles directement. Par contre, il leur est possible de communiquer via un mécanisme de messages [Fig.4]. La première partie sera essentiellement constituée de pages HTML. Celles-ci peuvent bien sûr contenir du Javascript.

Parmi ces pages, il est possible d'en définir une qui sera systématiquement chargée lorsque Safari chargera l'extension, mais qui ne sera pas affichée. Cette page est conventionnellement nommée global.html et est l'alter-ego du background.html rencontré avec Google Chrome.

L'autre partie contiendra des documents CSS ou Javascript destinés à être injectés dans les pages visitées. Il n'y a pas obligation à ce que ces deux parties soient non vides. Il est parfaitement possible qu'une extension rudimentaire soit constituée uniquement de la première, ou de la seconde partie. Enfin, contrairement à ce qui se passe avec Opera, le code Javascript d'une extension Safari ne partage pas l'espace d'exécution du code Javascript contenu dans les pages visitées.



## 5 HELLO WORLD!

Avec Safari les choses sont beaucoup moins immédiates qu'avec Opera, même pour un simple Hello World. Avant tout, vous devez vous inscrire au programme développeur d'Apple. Cela vous donnera l'indispensable accès la documentation, mais aussi la possibilité de générer un certificat sans lequel il est impossible de développer une extension et encore moins de la déployer. Vous devez donc avant toute chose vous rendre à <http://developer.apple.com/safari/>. Allez-y avec le navigateur Safari, cela vous évitera quelques problèmes d'affichage :) Une fois sur la page, repérez en haut de la colonne de droite, le pavé 'Safari Developer Program', et cliquez sur 'join now'. Une fois la (très lourde) procédure d'inscription terminée, retournez sur la même page, connectez-vous avec votre login et vous verrez dans le même pavé qu'il est possible d'obtenir ce fameux certificat en cliquant sur le lien 'get started'. Suivez la procédure et installez ce certificat.

## 6 LE CONSTRUCTEUR D'EXTENSION

Une des agréables originalités de Safari réside dans son constructeur d'extensions, ou 'Extension Builder' que nous nommerons simplement Builder par commodité. Cet outil centralise les informations relatives à votre extension, permet de définir quels fichiers HTML et Javascript seront attribués à telle ou telle tâche, de définir sur quelle(s) page(s) visitée(s) l'extension sera active, de créer le paquet de l'extension, de l'installer, etc. A partir de Safari 5.0.1 les extensions sont activées par défaut, et il est donc possible de faire apparaître le Builder depuis le menu 'Développement' du navigateur, ce que vous devez faire pour construire notre première extension Hello World. Au départ le Builder est désespérément vide. Cliquez sur le signe '+' en bas à gauche et sélectionnez 'Nouvelle extension'. On vous demande alors de donner le nom d'un répertoire auquel sera systématiquement ajouté '.safarixtension'. Dans ce répertoire, vous déposerez tous vos fichiers HTML, CSS et Javascript. Vous pouvez aussi créer des sous-répertoires dans lesquels vous déposerez des ressources. Des images par exemple. Si vous déposez un fichier icon.png à la racine, le Builder le prendra comme l'icône de votre extension. Plus généralement le Builder tient automatiquement compte de tout ce que vous déposez dans le répertoire. L'illustration montre notre extension à travers ce fameux Builder [Fig.5]. Pour notre Hello World, nous avons choisi de définir une nouvelle barre dans le navigateur. Un fichier HTML suffit pour cela. Nous avons donc déposé le fichier helloworld.html dans le répertoire. Voici le contenu du fichier :

```
<!DOCTYPE html>
<html>
<head>
<title>Hello World Programmez!</title>
</head>
<body>
Hello World Programmez!</body>
<!-- ou src="./images/logo-prog.png" -->
</html>
```

Dans la barre nous affichons donc un texte et une image qui est déposée dans un sous-répertoire. Remarquons que le tentant :

```

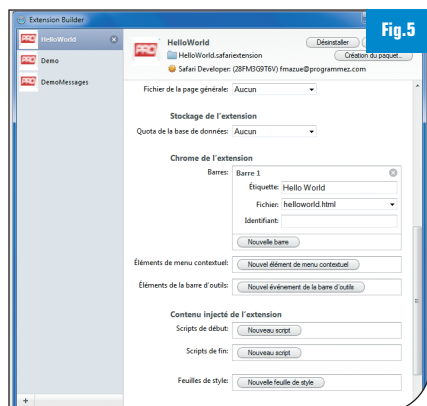
```

ne fonctionnerait pas et Safari ne chargerait pas l'image. Revenons au Builder. Comme nous définissons une nouvelle barre, il convient de cliquer sur le bouton 'Nouvelle barre'. A ce moment le Builder vous demandera quel est le fichier HTML correspondant. Hello-world.html fera partie de la liste, il suffit de le sélectionner. Tout est prêt. Il suffit maintenant de cliquer sur le bouton 'Installer', en haut et à droite du Builder, pour que l'extension soit opérationnelle dans Safari, comme illustré [Fig.6]. Si, au cours du développement d'une extension, il est nécessaire de modifier le contenu d'un fichier, alors on cliquera sur le bouton 'Recharger', qui est venu remplacer le bouton 'Installer' dans le Builder, afin que les modifications soient prises en compte. Remarquons que notre extension Hello World ne comporte qu'une seule partie, sur les deux théoriquement possibles.

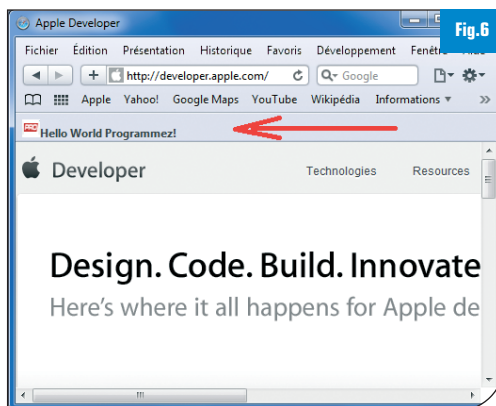
## 7 AGIR SUR LE DOM

Nous allons maintenant, comme nous l'avons fait avec Opera, modifier le DOM de la page d'accueil du site de Programmez!. Mais cette fois, puisque le script d'extension s'exécute dans son espace propre, il n'est pas immédiat de connaître le nombre de sections Javascript dans le document. Il faudrait analyser celui-ci, ce que nous laissons de côté car cela nous écarterait du sujet. Nous faisons donc une modification toute simple, ainsi qu'illustré [Fig.7]. Pour parvenir à cela nous allons créer une nouvelle extension, baptisée Demo, qui sera une extension d'injection de script seulement, et comme telle ne comportera également qu'une seule partie. Et dans cette partie un seul script extrêmement simple :

```
div_elem = document.getElementById('actus_accueil');
div_elem.innerHTML = 'Un bon conseil: ABONNEZ VOUS ! :-)';
```



Notre Hello World vu à travers le Builder de Safari.



Voici une nouvelle barre intégrée à Safari.



Notre seconde extension Safari agit sur le DOM de la page d'accueil de Programmez!

Ce script est bien entendu déposé dans le répertoire de l'extension. Maintenant tout se tient dans le Builder. Dans la section 'Contenu injecté de l'extension', on cliquera sur le bouton 'Nouveau script', puis on sélectionnera demo.js, tout simplement, et comme illustré [Fig.8]. Le script doit être un 'Script de fin' ce qui signifie qu'il sera exécuté une fois la page visitée chargée. A contrario, un 'Script de début' est exécuté avant le chargement de la page. Maintenant la question est de savoir comment faire pour que le script soit injecté dans la page d'accueil de Programmez!, dans cette page seulement sur notre site, et sur aucune autre page d'autres sites. A nouveau, ceci s'obtient dans le Builder. Tout d'abord, dans la section 'Accès au site web de l'extension', on sélectionnera 'Certaines' comme niveau d'accès, puis on cliquera sur le bouton 'Nouveau motif de domaine', et l'on renseignera avec :

www.programmez.com

comme illustré [Fig.9]. Bien remarquer l'absence du protocole http://. A ce stade l'extension ne sera pas active sur d'autres sites. Maintenant, il suffit d'ajouter la page d'accueil dans la liste blanche et cette page seulement. Ceci est visible sur l'illustration qui montre la définition du script [Fig.10]. On donne donc :

http://www.programmez.com/

Bien remarquer la présence du / final, et, surtout, la présence du protocole http://.

## 8 LE MÉCANISME DE MESSAGES

Pour notre dernier exemple, nous allons construire une application en deux parties, que nous baptisons DemoMessages, afin de nous familiariser avec le mécanisme de messages qui permet la communication entre les deux parties de l'extension. Notre extension reste une injection de script, mais nous partons du principe que le script doit demander à ce pivot qu'est la page invisible global.html, quel texte doit être injecté dans la page visitée. Voici le code de global.html

```
<!DOCTYPE HTML>
<html>
<head>
<title>global HTML page</title>
<script type="text/javascript">

var msg = "Un bon conseil: ABONNEZ-VOUS! :-)";
```

```
function repondeur(theMessageEvent) {
    if(theMessageEvent.name === "conseil")
    {
        theMessageEvent.target.page.dispatchMessage("reponse-conseil", msg);
    }
}
safari.application.addEventListener("message", repondeur, false);
</script>
</head>
<body>
</body>
</html>
```

Le fichier global.html étant déposé dans le répertoire de l'extension, la section 'Page générale de l'extension' du Builder doit être renseignée avec lui. Le code est très simple. On définit un écouteur qui, lorsqu'il est sollicité, répond au message qui lui est envoyé. Voyons maintenant le code du script d'injection :

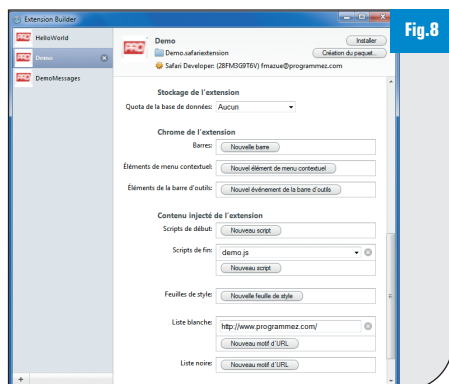
```
function demander() {
    safari.self.tab.dispatchMessage("conseil");
}

function recevoir(theMessageEvent) {
    if (theMessageEvent.name === "reponse-conseil") {
        div_elem = document.getElementById('actus_accueil');
        div_elem.innerHTML = theMessageEvent.message;
    }
}

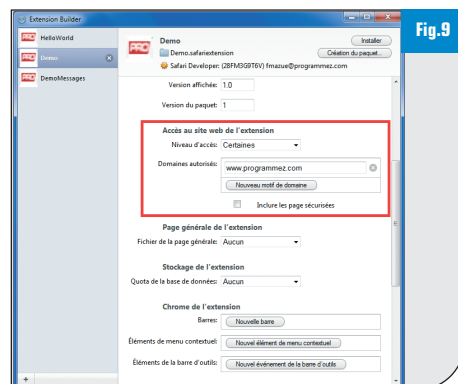
safari.self.addEventListener("message", recevoir, false);
demander();
```

La fonction 'demander' envoie un message à la page globale. Nous savons que celle-ci va répondre. Mais comme elle le fait par un message elle aussi, il faut être capable de capter cette réponse. C'est pourquoi notre code définit un écouteur à cet effet, avant d'émettre sa demande à la page globale. Ce mécanisme est simple. Le lecteur veillera toutefois à éviter les confusions entre tab, page, target et tabs[], lorsqu'il s'agit d'invoquer dispatchMessage, car une erreur difficile à corriger par la suite est vite arrivée. Nous le renvoyons pour cela à la documentation qui explique les choses relativement clairement.

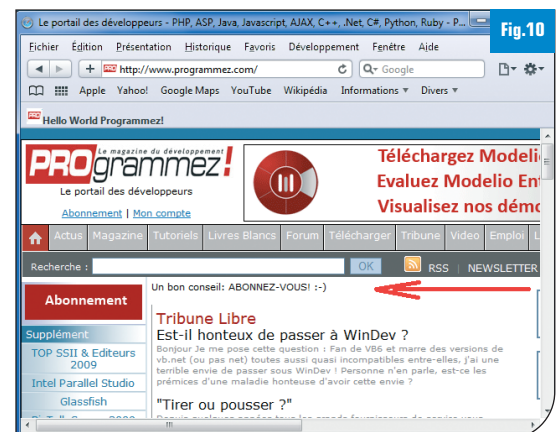
■ Frédéric Mazué - [fmazue@programmez.com](mailto:fmazue@programmez.com)

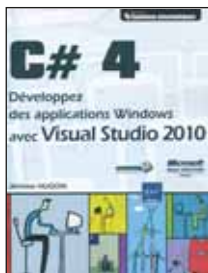


Définition d'une extension d'injection de script sans le Builder de Safari.



Notre extension Demo ne doit être active que sur le domaine www.programmez.com.





## LANGAGE

### C# 4

Difficulté : \*\*\*

Editeur : éditions Eni

Auteur : Jérôme Hugon

Prix : 39€

Ce livre se consacre au développement Windows et .Net autour du langage C# 4.0 et VS 2010. L'auteur a choisi une approche pas à pas tout en construisant une application fonctionnelle tout au long de l'ouvrage pour illustrer de manière pratique et cohérente les concepts abordés. L'apprentissage commence par la familiarisation avec l'interface de Visual Studio 2010 et le concept de l'architecture .NET. Les détails du langage C# (en version 4.0 au moment de la rédaction du livre), sa syntaxe et ses fonctionnalités comme les classes, l'héritage, les interfaces, les types génériques ou encore les délégués et les événements sont ensuite expliqués avant d'aborder la conception d'interfaces utilisateur. Des concepts plus avancés sont également traités, afin d'exposer une gamme plus large des possibilités offertes par le langage C# et Visual Studio : l'utilisation des expressions régulières, le développement d'applications multi-tâches, la globalisation et la localisation d'une application, la sécurité du code, l'implémentation d'applications client/serveur, le dessin avec GDI+ ainsi que la réflexion font partie des sujets abordés. Une bonne introduction progressive à C# 4.

## LANGAGE

### Algorithmique en C



Difficulté : \*\*\*\*

Editeur : Pearson

Auteur : J-M Léry

Prix : 19 €

On ne le répètera jamais assez, l'algorithmique est un passage obligé et essentiel de la programmation. Cet ouvrage présente comment concevoir de bons algorithmes, comment les optimiser, en langage C. Mais ces bases peuvent s'appliquer à d'autres langages. L'auteur propose des exercices pour vous entraîner rapidement aussi bien en pseudo-langage qu'en langage C. La première partie porte sur la conception d'algorithmes. Elle expose les mécanismes tels que les tests et les boucles, ainsi que les

## SPÉCIAL LINUX

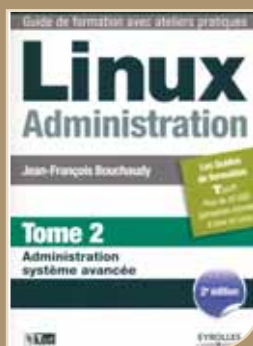
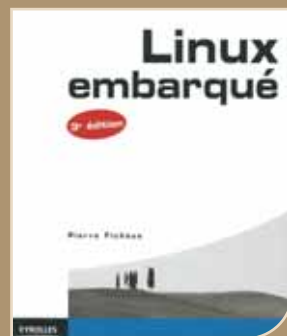
### Linux embarqué, 3<sup>e</sup> édition

Difficulté : \*\*\* - Editeur : Eyrolles

Auteur : Pierre Fichoux - Prix : 39 €

Les plateformes embarquées se multiplient et de nombreux systèmes sont disponibles sur ces environnements qui vont des caisses de magasins aux boîtiers de télévision en passant par les box ADSL.

Le système Linux constitue une des références de ce marché. L'auteur dresse, dans cette nouvelle édition, un panorama du marché et explique comment fonctionne un Linux embarqué, sans omettre la partie matériel. Vous trouverez même la manière de créer pas à pas une distribution embarquée. Instructif et très clair, à lire absolument !



### Linux administration tome 2, 2<sup>e</sup> édition

Difficulté : \*\*\*\* Editeur : Eyrolles

Auteur : J-F Bouchaudy - Prix : 25 €

Le premier tome de Linux administration servait à établir les bases de connaissance autour de Linux. Dans ce 2<sup>e</sup> volet, l'auteur va bien plus loin : Ext4, création d'une clé USB bootable, LVM, utilisation d'un San... Vous pousserez à l'extrême votre Linux et ses possibilités. Cette « formation » se déroule autour de fiches réparties par module. Le texte est très clair, très direct avec des cas concrets d'utilisation, des schémas. Si vous avez besoin d'administrer un Linux, l'investissement est largement rentabilisé !

structures de données comme les tableaux et les listes chaînées, qui servent de support à de nombreux algorithmes. La seconde partie étudie les algorithmes existants et leur utilisation. Elle aborde également les modèles de données couramment employés comme les piles, les files, les arbres, et traite des algorithmes récurrents.

## MONDE

### Révolution numérique dans les pays en développement



Difficulté : \*\*

Editeur : Dunod

Auteur : Jacques Bonjawo

Prix : 19 €

Les nouvelles technologies de l'information submergent le monde depuis un demi-siècle et tout spécialement les « pays riches » mais depuis 15-20 ans, cette révolution numérique touche les pays dits en développement et l'Inde fut l'un des précurseurs en la matière. Car ces technologies représentent une opportunité pour ces pays. Riche d'exemples concrets et de l'expérience de son auteur, cet ouvrage dresse un panorama des nombreuses possibilités offertes par les TIC et de leurs contributions à l'essor des pays en développement, notamment en

## LANGAGE

### Programmer en Erlang

Difficulté : \*\*\*

Editeur : Pearson

Auteur : collectif

Prix : 47 €



On ne présente plus le langage fonctionnel Erlang. Bien que trop souvent méconnu, Erlang possède une puissance inégalée aussi bien en multi-core qu'en fiabilité. Vous y découvrirez la force d'Erlang et explorerez les concepts liés à la concurrence et la façon dont il la gère.

Vous apprendrez à écrire des programmes efficaces dans un code propre et lisible, à ajouter des interfaces graphiques simples avec peu d'effort, à connaître les mécanismes de trace du langage pour déboguer... Il est écrit par deux acteurs actifs de la communauté Erlang.



Lorsque des experts français en **développement .NET** s'associent  
avec des vétérans de la **formation Microsoft** aux États-Unis



&



**pluralsight**  
see what you can learn

## «Le meilleur des formations Microsoft en France»



Les cours sur toutes les technologies de développement  
Microsoft qui ont fait la réputation de **Pluralsight** assurés  
par des experts locaux avec de vraies expériences du terrain

La possibilité de parcours pédagogiques sur mesure combinant  
de manière optimale, formation sur site et formation en ligne



L'accès 24/24 à une bibliothèque unique de cours en ligne  
sur la plate-forme **.NET**, distribuée en France exclusivement  
par **SoftFluent**

*Pour obtenir un mois d'accès gratuit  
à la bibliothèque, écrivez dès à présent à*

**[training@softfluent.com](mailto:training@softfluent.com)**

# L'INFINI EST MA SEULE LIMITE DE DEVELOPPEMENT. C'EST ÇA LA PUISSANCE DU CLOUD.



Pour utiliser ce flashcode,  
téléchargez l'application  
gratuite sur :  
<http://gettag.mobi>

Windows Azure est une plateforme qui vous permet de déployer et exploiter vos applications dans le Cloud avec vos outils et compétences d'aujourd'hui. En outre, c'est une plateforme d'une élasticité infinie qui épouse parfaitement les fluctuations de votre activité. Avec Windows Azure, pensez à votre potentiel avant de penser à vos contraintes ; et n'ayez comme seule limite que votre imagination.

La simplicité - une des nombreuses vertus du Cloud.

**Microsoft**



**Cloud Power**

**LA PLUS LARGE GAMME DE SOLUTIONS DANS LE NUAGE. SUR TERRE.**

Microsoft® Office 365 • Windows Azure™ • Windows Server® Hyper-V® • Microsoft Dynamics™ CRM Online.

En savoir plus : [www.microsoft.fr/cloud](http://www.microsoft.fr/cloud)