

# PROgrammez!

Le magazine du développement

www.programmez.com



## La Révolution du MULTI-TOUCH

Comment ça marche ?  
Les outils, les technologies disponibles  
Coder pour l'iPhone 2.0  
Au cœur du e-lab de Bouygues

 Microsoft®  
**SQL Server® 2008**

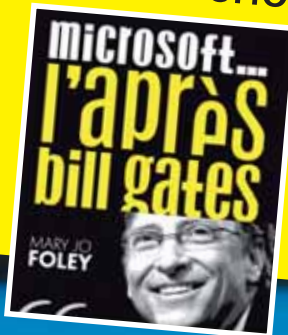
300 licences  
À GAGNER

## ENFIN DISPONIBLE !

**Google**  
Intégrer Google Maps et JSF  
**Mozilla**  
Créer votre extension XUL  
pour Firefox !

**Multicœur**  
■ Programmer vos cœurs  
avec OpenMP  
■ La charge processeur  
avec WMI

**EXCLUSIF**  
Les révélations  
du livre choc



L'équipe  
The Shads,  
de l'Epita  
ABC Project

## Les Jeux Olympiques du développement !

*ImagineCup révèle  
les nouveaux talents*

## Spécial PHP

- Les nouveautés PHP 5.3
- Créer son moteur de template en PHP 5
- UML et PHP : le duo d'enfer
- PHP et la programmation par composants

M 04319 - 111 - F: 5,95 €



VERSION  
EXPRESS  
GRATUITE

Téléchargez-la !

Environnement intégré (IDE & ALM)

## INCLUS EN STANDARD DANS WINDEV 12 :

- Tout est en **français** (manuels, logiciel, exemples)
- Hot Line personnalisée (**support technique**) gratuite\*
- Déploiement **libre et gratuit**
- Crée des **.EXE** sécurisés, des **WebServices**, des applications **.NET**, des applications **Java...**
- Fonctionne en **TSE** et Citrix
- 32 bits, **64 bits**
- **Compilation JIT**
- **Code multi-plateforme compatible Mobile et Internet**
- Générateur automatique d'**IHM**, avec charte graphique. Création automatique de superbes fenêtres par utilisation de gabarits fournis (pour tous Windows)
- **Tous les Windows** : 98, 2000, NT, XP, Vista, 2008...
- Générateur d'**Etats** et de **Requêtes diffusables**, création de **PDF**, code-barres, étiquettes. Fond de page PDF
- **Menu d'export automatique** dans chaque application créée: vers Word, Excel, OpenOffice, XML, PDF; Graphiques 3D; Historique de saisie,... Envoi de mail, Macros
- **Real-RAD** : Générateur d'applications complètes, avec possibilité de créer ses propres Patterns
- **HyperFileSQL**, Base de Données Client/Serveur, Locale et Mobile sous Windows et Linux (version illimitée, **libre** et gratuite); Gère 4 millions de Téra-octets
- Accès à toutes les Bases de Données tierces: **Oracle, AS/400, SQL Server, DB2, MySQL, Access, xBase**, etc...
- **Réplication** entre bases de données
- **XML** natif
- Accès natif à **SAP R/3**, Lotus **Notes**, LDAP, Outlook,...
- Centre de Modélisation **UML**, Merise et Souple; code généré depuis l'analyse, reverse engineering
- Centre de suivi du **planning** d'équipes
- Tableau de Bord de **suivi de projet**
- **Dossier** automatique : analyse & programmation
- Règles métier; **Gestion des exigences**
- Création et utilisation de **composants** ; 3-tiers
- Centre de suivi des retours et suggestions utilisateurs
- Import d'applications **Access** et **VB**
- Langage de 5<sup>e</sup> Génération **L5G**, qui élimine 90% du code
- **Ouverture** aux L4G et L3G: C++, C#, Java, VB, Cobol...
- Fonctions **Domotique** (norme X10)
- Gestion **liaison série** RS 232, parallèle et **USB**
- Fonctions Bluetooth,
- Fonctions réseau **SNMP**
- Fonctions TAPI, OPC, FTP, HTTP, Socket, Twain, API, DLL,...
- Fonctions Multimédia (image, son, vidéo)
- Editeur de code intelligent, avec test immédiat sans recompilation
- Gestion des **versions** (gestion des sources: en local, à distance)
- Automate de **tests unitaires** de code et d'IHM, Editeur visuel de tests de non-régression
- Débogueur puissant: threads, composants,...
- **Débogage à distance**
- Refactoring
- **Profiler**, pour optimiser la vitesse du code
- Multilingue automatique: jusqu'à 20 langues
- Générateur d'aide CHM
- Générateur d'**Installations en 1 clic**, gérant les mises à jour automatiquement (local, à distance, via Internet)
- Autoformation facile, en 1 semaine (**manuels livrés**)

## DÉVELOPPEZ 10 FOIS PLUS VITE

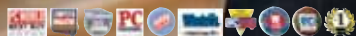
Développez  
facilement de  
superbes  
applications  
robustes et  
rapides avec le  
N°1 en France.



[www.pcsoft.fr](http://www.pcsoft.fr)

Demandez le guide gratuit (244 pages + 1 DVD)  
**VERSION EXPRESS GRATUITE** et 112 Témoignages.  
Tél: 04.67.032.032 ou 01.48.01.48.88 Mail: [info@pcsoft.fr](mailto:info@pcsoft.fr)

Fournisseur Officiel  
Préparation Olympique





# sommaire

## \\ actus

L'actualité en bref .....	6
Agenda .....	10

## \\ événements

PHP 5.3 : toutes les nouveautés .....	12
---------------------------------------	----

## \\ gros plan : SQL Server 2008

Tout savoir sur SQL Server 2008 .....	18
SQL Server 2008 : une offre étudiée .....	21

## \\ dossier : Multi-Touch

La révolution du Multi-Touch .....	24
Comment ça marche .....	26
Développez pour l'iPhone .....	30
Les nouvelles IHM à base d'interface tactile .....	35

## \\ reportage : ImagineCup 2008

Les jeux Olympiques du développement .....	38
Les Français : au sommet d'ImagineCup2008 .....	40
Imaginer le caddie écologique de demain : le défi de The Shads .....	44
Réseau : Médaille d'or pour Jean-Benoît Paux .....	45
Projet Soak : économiser 50% de l'eau du pays .....	46

## \\ bonnes feuilles

Microsoft... l'après Bill Gates .....	49
---------------------------------------	----

## \\ code

Un moteur de template en PHP 5 .....	53
Développer à l'aide de composants .....	56
UML et le génie logiciel : développer avec PHiMX .....	58
Combiner Google Maps et composants JSF .....	61
Flex 3 compatible Linux ? .....	64
Créer une extension Firefox avec XUL .....	65
OpenMP et les processeurs multi-cœurs .....	68
Observez la charge de votre processeur multi-cœur sous Windows .....	72
SQLAlchemy : faites du mapping objet/relationnel en Python .....	79

## \\ temps libre

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

16



24



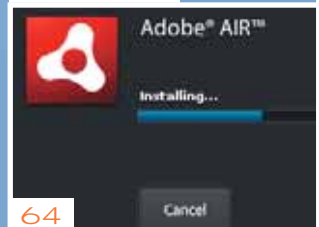
38



46



49



64

# CD ROM

### Jeu

**Ecological Tycoon**  
Exclusif : le jeu de stratégie à la Sim City, finaliste à Imagine Cup + les librairies DirectX 9, XNA Framework, Framework .Net 2.0

### PHP

**PHP 5.3**  
Découvrez dès aujourd'hui les nouvelles fonctions du langage web de référence !  
**Windows**  
**PHP 6.0 alpha**  
La prochaine version majeure de PHP avec support d'Unicode. *Windows*

### PHiMX

Outil de génération de code et de modélisation UML, pour les projets PHP 5. Open Source.

### Java

**Eclipse Classic 3.4 SDK**  
Développez avec la dernière version d'Eclipse.

**Sling Standalone Application**  
framework Web pour le développement web en Java

### UML

**StarUML 5.0**  
Modèleur UML Open Source, écrit en Delphi. *Windows*

**Umbrello UML Modeller 1.5.8**  
Modèleur UML pour KDE

### Outils

**CodeFluent 1.6**  
Fabrique logicielle entièrement pilotée par les modèles, pour .NET. *Windows - Evaluation, 60 jours*  
**Allcapture**  
ALLcapture enregistre les activités de votre écran en temps réel. *Windows*  
**TurboDemo 7.5**  
**WSO2 Mashup Server 1.5**  
Plate-forme de création et déploiement de Mashups  
**Subversion 1.4.6**  
Le gestionnaire de version qui se veut le successeur de CVS.

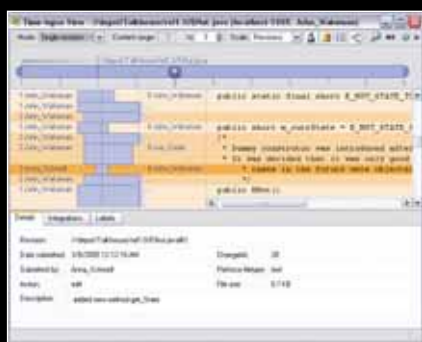
Supplément éditorial n°111

<b>PHP 5.3</b> Découvrez dès aujourd'hui les nouvelles fonctions du langage web de référence ! <b>PHP 6.0 alpha</b> La prochaine version majeure de PHP avec support d'Unicode. <b>PHiMX</b> Outil de génération de code et de modélisation UML, pour PHP 5. <b>Net</b> <b>CodeFluent 1.6</b> Fabrique logicielle entièrement pilotée par les modèles. <i>Windows Evaluation, 60 jours</i> <b>UML</b> <b>StarUML 5.0</b> Modèleur UML, Open Source, écrit en Delphi. <b>Umbrello UML Modeller 1.5.8</b> Modèleur UML pour KDE	<b>JEUX</b> <b>Ecological Tycoon</b> Exclusif : le jeu de stratégie à la Sim City, finaliste à Imagine Cup + les librairies DirectX 9, XNA Framework, Framework .Net 2.0 <b>JAVA</b> <b>Eclipse Classic 3.4 SDK</b> Développez avec la dernière version ! <i>Windows</i> <b>Sling Standalone Application</b> Le nouveau Framework pour le développement web en Java <b>OUTILS</b> <b>WSO2 Mashup Server 1.5</b> Plate-forme de création et déploiement de Mashups <b>Allcapture</b> : Enregistre les activités de votre écran en temps réel et crée des vidéos à publier sur internet - <b>TurboDemo 7.5</b> : Capture d'écran avec déplacements de la souris, éditeur sous forme de dispositifs.
--	--



## Présentation de la fonctionnalité Time-lapse View,

**un atout de productivité du système de GCL Perforce.**



Fonctionnalité Time-lapse View du logiciel Perforce

La fenêtre d'évolution temporelle (Time-lapse View) permet aux développeurs de voir chaque modification apportée à un fichier. Cette fenêtre offre une présentation dynamique sous forme d'annotations. Les développeurs peuvent enfin obtenir des réponses à leurs questions, notamment : « Qui a écrit ce code et quand ? » et « Quelles sont les modifications apportées au contenu et qu'est-ce qui justifie ces changements ? ».

L'outil Time-lapse View se présente sous la forme d'une ligne de temps qui rend compte, de manière claire, de l'évolution d'un fichier, modification après modification. Les dégradés de couleur reflètent l'âge du contenu du fichier et la ligne de temps peut être configurée pour indiquer les modifications apportées au fichier par numéro de révision, par date ou par numéro de modifications.

La fonctionnalité Time-lapse View n'est qu'un des nombreux atouts de productivité offerts par le système de GCL Perforce.



## C'est grave, docteur, la "versionite aiguë" ?



Avant, tout était simple : une version, des patches, des mises à jour puis une nouvelle version majeure, etc. Aujourd'hui, on nous propose des pré-versions, de pré-pré-versions, des bêtas ad vitam aeternam ! De quoi s'interroger sur l'intérêt de prendre tel millésime d'un logiciel. Cette versionite aiguë est stratégique pour les éditeurs, mais il faut la manier avec prudence. Les développeurs doivent tout de même utiliser la nouvelle version sans attendre la suivante, disponible en alpha ou bêta. Et c'est où le bât blesse ! Sur ce point, Microsoft a fait de beaux impairs : présentation de Silverlight 1.0 et 1.1 (renommée 2.0), idem pour la gamme Expression en dévoilant les pré-versions de Blend 2.0 et 2.5 quasi simultanément. Comment ne pas frustrer notre ami développeur en disant : " n'attendez pas 6 mois même si dans 6 mois, la prochaine version est *plus mieux* ! ". Confusion dans les esprits, frustrations, amalgames...

Que dire sur le trio d'enfer : XP, Vista, Seven ? Les uns restent en XP en attendant, finalement Seven, mais pour l'éditeur, il faut passer à Vista, sans attendre... C'est aussi la question que l'on se posait sur les mises à jour majeures constantes des distributions Linux, tous les 6-7 mois. C'est très bien mais cela signifie refaire une batterie de tests, de validations... Le développeur n'a pas à perdre des heures !

Mais notre versionite aiguë est aussi due au Web 2, à Google, à la guerre des services en ligne, hébergés et autres SaaS, S+S, etc. Aujourd'hui, on vilipende un éditeur avec ses mises à jour constantes, mais sur le web, cela paraît presque normal de rester en bêta plusieurs années. Manière aussi de dire, " *Ah ben non, on ne peut pas me reprocher de n'avoir pas cette API, ce protocole, je suis en bêta* ". L'excuse est belle. Avec pour l'utilisateur, le développeur une instabilité dans les fonctions, les API et nécessitant parfois une réécriture non négligeable des applications au changement de bêta. L'un des défis des éditeurs en ligne est d'éviter de casser l'existant. Et il faut dire aussi que le contexte technologique du Cloud Computing est tellement balbutiant que tout le monde se cherche un peu, tente des solutions. Et que dire des environnements qui se mettent aux services hébergés, comment un DBA va suivre les évolutions car il perdra la main sur l'administration des versions, des patches... Idem pour le développeur qui sera condamné à suivre les évolutions en ligne. Et cette dictature de la version n'est pas une bonne chose.

Bref après le vaporware, nous sommes rentrés dans le merveilleux monde du versionware... Et soyez content, de toute manière, vous n'avez pas le choix !

■ FRANÇOIS TONIC

## COUP DE JEUNE



Alors que les Jeux Olympiques se sont clôturés récemment, une autre manifestation se tenait à Paris, fin juin, confrontant l'élite mondiale des jeunes développeurs : Imagine Cup. Les 370 champions - tous étudiants - venant de 61 pays, étaient sélectionnés parmi 200 000 candidats !

Bien sûr, la compétition est organisée par Microsoft, sur la base de ses technologies. Il s'agit donc aussi d'un formidable outil de promotion pour la marque. Mais les Jeux Olympiques ne sont-ils pas aussi une vitrine publicitaire pour les nations et les marques sponsors ?

Et quel beau thème, cette année, que celui de ces olympiades de l'intelligence : " Imaginez un monde où la technologie permet un environnement durable " ! Développement ... durable ! Les deux termes accolés prennent ici tout leur sens. Oui, l'informatique change la société et contribue à façonner un nouveau monde. Des jeunes (18-20 ans) s'enthousiasment pour cet outil, loin de penser que programmer, c'est seulement " pisser du code " ! Quelle belle occasion pour marquer cette nouvelle formule de votre magazine. Certes, une maquette renouvelée, un logo " rajeuni " mais aussi, sur le fond, davantage de pages et de portraits sur le thème des carrières, de la vie professionnelle, des études et formations.

■ JEAN KAMINSKY

Rédaction : redaction@programmez.com  
 Directeur de la Rédaction : Jean Kaminsky  
 Rédacteur en Chef : François Tonic  
 Ont collaboré : F. Mazué, L. Guallois, F. Dewasmes, N. Toukiss, C. Remy, .  
 Experts : JP Riehl, D Seguy, J Pauli, E Vautherin, J. Petitcolas, A. Vannieuwenhuyze, F. Tribble,  
 Illustrations couverture : Microsoft. Jean Kaminsky  
 Maquette : AJE Conseils  
 Publicité : Régie publicitaire, K-Now sarl  
 Pour la publicité uniquement : Tél. : 01 41 77 16 03  
 coordination@programmez.com  
 Éditeur : Go-02 sarl, 6 rue Bezout - 75014 Paris  
 Coordination@programmez.com - Dépôt légal :  
 à parution - Commission paritaire : 0707K78366  
 ISSN : 1627-0908 - Imprimeur : ETC - 76198 Yvetot  
 Directeur de la publication : J-C Vaudecrane  
 Ce numéro comporte 1 CD Rom et un encart  
 Component Source sur une partie du tirage.

Abonnement : Programmez 22, rue René Boulanger, 75472 Paris Cedex 10 - abonnements.programmez@groupe-gli.com Tél. : 01 55 56 70 55 - Fax : 01 55 56 70 20 - 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 : 45 € - Etudiant : 39 € - CEE et Suisse : 51,83 € Algérie, Maroc, Tunisie : 55,95 € Canada : 64,33 € Tom : 79,61 € - Dom : 62,84 € - Autres pays : nous consulter.  
 PDF : 30 € (Monde Entier) souscription en ligne.

L'INFO  
 PERMANENTE  
 WWW.PROGRAMMEZ.COM



## PROCHAIN NUMÉRO

N°112 octobre, parution 30 septembre

### • Spécial 10<sup>e</sup> anniversaire !

Programmez! a 10 ans, occasion de revenir sur les 10 dernières années du développement et le futur de notre métier. Découvrez les résultats de notre grande enquête.

### • Eclipse 3.4 Netbeans 6.1 : lequel choisir ?

Netbeans a-t-il encore un avenir ? Eclipse 4.0 : le futur du développement Java ?







## Information Management

### \_LE JOURNAL DE NOTRE INFRASTRUCTURE

\_94<sup>e</sup> JOUR : Impossible de retrouver les données clients dont nous avons besoin. Comment exploiter nos informations si elles ne sont pas facilement accessibles?

\_Gilles a installé une machine à téléporter : il prétend qu'il peut transférer les gens instantanément dans leur espace de données... permettant au marketing de se libérer des données discordantes et hétéroclites.

\_97<sup>e</sup> JOUR : La réponse est ailleurs. Le logiciel IBM InfoSphere Master Data Management Server nous apporte une vue unifiée de nos informations clients en temps réel. Nous alimentons les utilisateurs, les processus et les applications avec des données fiables. Et nous obtenons de meilleurs résultats pour l'entreprise.

\_Aïe ! Je viens de téléporter Gilles en Espagne. ¡Lo siento, mi amigo !\*

Pour accéder rapidement à une information fiable :  
[IBM.COM/TAKEBACKCONTROL/DATA/FR](http://IBM.COM/TAKEBACKCONTROL/DATA/FR)

#### ■ Mozilla Labs : quel futur ?

Pour le futur du web, Mozilla Labs lance " Concept Series " autour des projets développés en interne tels que Aurora, Mobile Concept. Quel sera votre concept ? C'est le défi que Mozilla vous lance ! A vous d'imaginer, de créer et de proposer. Site :

<http://labs.mozilla.com/projects/concept-series/>

#### ■ IBM veut Ilog

IBM a annoncé cet été sa volonté de racheter le français ILOG, spécialisé dans les outils de règles métiers, les composants graphiques pour Flex, .Net et Java. Ce rachat permettra à IBM de compléter sa gamme logicielle qui ne possédait pas d'offres sur ces deux domaines.

#### ■ Un nouveau rendez-vous libre à Paris

Les 1er et 2 décembre prochains se tiendra à Paris le Forum Mondial du Libre. La volonté est de présenter toute la diversité du monde libre. Aucun détail à l'heure où nous écrivons. Site :

<http://www.ow2.org/view/Events/ForumMondialduLibre>

#### ■ PHP 4 est mort

Depuis le 8 août dernier, PHP 4 n'est plus supporté et officialise ainsi la mort de cette lignée au profit de PHP 5.x dont la v5.3 s'annonce particulièrement prometteuse.

## Outils

# Netbeans 6.5 déjà là !

La première bêta de Netbeans 6.5 a été dévoilée mi-août dernier. Cette version s'appuie naturellement sur les fondations de la 6.1, de Java EE 5 et de Glassfish 2. La partie PHP va connaître diverses modifications sur l'éditeur, le debug (avec Xdebug). Sur javascript / Ajax, on aura son lot d'améliorations (éditeur, debug, support des diverses librairies comme Yahoo UI,



Dojo, Prototype, un éditeur CSS revu et corrigé). Groovy et Grails connaîtront aussi des modifications sur la génération, l'éditeur et le Framework rails. Ruby et Rails suivent le même chemin dans une nouvelle interface de test. Côté Java EE, Glassfish 3 sera pré-

sent en preview. La partie mobilité subit aussi quelques améliorations comme un nouvel assistant, de nouveaux composants SVG, par contre, pas encore de JavaFX en standard. Cette version est annoncée en version finale pour mi-octobre prochain ! ■

## Outils

# Service Pack 1 pour Visual Studio 2008 et .Net 3.5

Après plusieurs mois de bêta test, Microsoft a lancé mi-août le premier service pack pour VS 2008 et .Net 3.5. Cette version rajoute de nombreuses fonctions, corrections de bugs, améliorations, ce qui va obliger les développeurs à tester et intégrer progressivement le SP1. Concernant VS 2008, ce service pack améliore l'éditeur WPF, apporte le support de SQL Server 2008 ainsi que d'un éditeur pour ADO.Net Entity. La partie C++ se complète d'un style Office 2007 et la partie

Ajax n'est pas oubliée ainsi que le déploiement des sites web. Le SP1 pour .Net 3.5 apporte des performances en hausse pour les applications WPF. Sur WCF, il améliore l'accès aux données et services. Il supporte dorénavant Entity, SQL Server 2008, Data Services. Team System n'est pas en reste avec quelques améliorations sur les performances, la montée en hausse ou encore la prise en compte des dernières versions des applications Microsoft. Les réactions ne se sont pas fait attendre. Ainsi, Cyril Durand (développeur ASP.Net et auteur dans Programmez !) a noté un curieux problème après publication d'un site en ASP.Net 3.5 SP1.

Venez en discuter sur nos forums ! ■

Blog de Cyril : <http://blogs.codes-sources.com/cyril/archive/2008/08/14/asp-net-3-5-sp1-could-not-load-type-system-web-ui-scriptreferencebase.aspx>

Pour en savoir plus sur ce SP1, Microsoft propose un .Net Framework 3.5 Enhancements Training Kit : <http://www.microsoft.com/downloads/details.aspx?FamilyID=355c80e9-fde0-4812-98b5-8a03f5874e96&displaylang=en>

## Conférence

# Paris capitale du libre

Les 24 et 25 septembre prochains, le monde du libre se réunit à Paris. Cette année encore, le programme sera particulièrement chargé avec de nombreuses conférences sur la situation du libre en France et dans le monde, les aspects entreprise, dans le secteur public, et bien entendu, les aspects business du marché open source. Les organisateurs espèrent beau-

coup de monde dans ces conférences (avec de nombreux intervenants) et autour des stands. Les Lutèce d'Or seront décernés à cette occasion. Ces trophées visent à apporter une reconnaissance à des projets, et auprès des communautés de développeurs, éditeurs, etc. et bien entendu, à favoriser et aider les initiatives. Cette année, ce n'est plus seulement la France qui sera présente, mais l'international. Attention : vous avez jusqu'au 5 septembre pour déposer votre candidature ! Programmez ! sera bien entendu présent durant tout l'événement dans les conférences et les Lutèce d'Or. ■

Site : <http://www.paris-libre.org>

## Exclusif !

Les coulisses de Microsoft sur [www.programmez.com](http://www.programmez.com)

Début septembre, une page inédite du livre «L'après Bill Gates» chaque jour, sur le site !



## Langage

# Google veut-il concurrencer XML ?

On connaissait l'ambition de Google dans la téléphonie avec le projet Android, maintenant c'est au tour de XML d'être dans la ligne de mire de l'éditeur. Début juillet, Google libérait dans la nature un projet interne utilisé pour fédérer l'ensemble des données et requêtes : Protocol Buffers. Pour faire simple, il s'agit d'un langage pour décrire et encoder des données (sérialisation automatique). Il se veut plus léger, plus compact, plus simple et plus performant que XML. On définit la structure voulue pour la donnée, puis on peut l'utiliser en lecture, écriture (via un code à écrire). Surtout, il permet d'utiliser diverses méthodes pour transporter la donnée et d'employer différents langages pour son utilisation. On peut même la mettre à jour sans changer les applications ou les données déjà déployées. On passe par des fichiers .proto. Chaque message Protocol Buffer est un petit "enregistrement logique" de l'information contenant des séries de valeurs de noms. Il indique clairement la structure, les champs, le type de données, etc. Ensuite, une fois le message défini, on exécute le compilateur adapté au langage de son application, qui génère alors le code d'accès aux données pour le langage voulu. Protocol Buffers est disponible pour les langages Java, C++ et Python. Pour l'occasion, Google a sorti la v2 et



non la v1. Cette version a été entièrement réécrite. Et il est vivement conseillé aux développeurs de créer de nouveaux supports langages car le choix est pour le moment limité. Moins verbeux que XML, il fournit aussi un code, un message dont la validation est plus facile à vérifier et à prouver qu'avec XML. Cependant, un gros point d'interrogation subsiste : quel avenir en dehors de Google pour ce projet ? C'est là la principale faiblesse de l'annonce. XML est aujourd'hui partout, utilisé par tous. Pour espérer prospérer, des éditeurs devront intégrer et supporter Protocol Buffers.

## Agenda

### Google parle aux développeurs français

Le 18 septembre se tiendra à Paris une journée développeur Google ! Plusieurs centaines de personnes sont attendues durant toute la journée. Il s'agira de faire le point sur l'ensemble des API et outils de l'éditeur, de venir rencontrer les responsables et développeurs internes, de parler code, intégration, évolution, business plan. Deux types de sessions sont prévues : session atelier pour écouter, discuter avec les ingénieurs, et les Code Labs pour coder ! La journée promet d'être chaude en informations et précisions. AppEngine, Android, OpenSocial et les Mashup tiendront la vedette ! L'événement se déroule dans le monde entier jusqu'en octobre prochain avec des centres d'intérêts différents selon les pays.

Site : <http://code.google.com/intl/fr/events/developer-day/2008/home.html>

*Mieux que DOM et SAX*

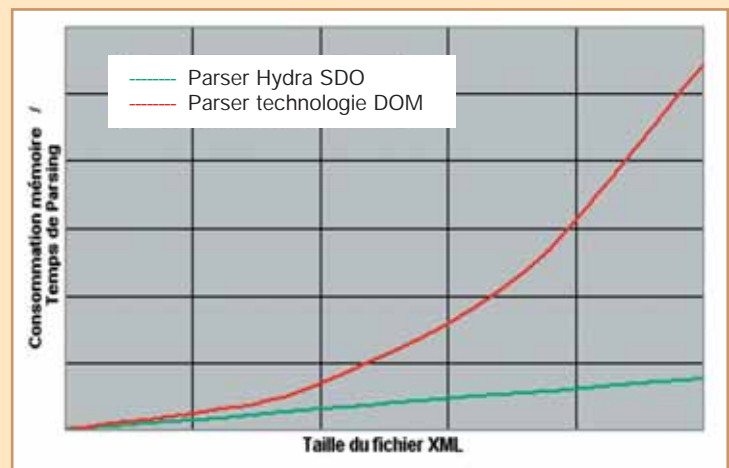
# Hydra SDO XML

Combine la souplesse de traitement apportée par DOM, avec la puissance de gestion des gros fichiers fournie par SAX

- Optimisé pour une faible consommation mémoire et des temps de réponse rapides, parse de très gros fichiers qui excèdent les capacités de la technologie DOM
- Conserve en mémoire l'ensemble des informations permettant toutes les opérations et traitements sur le contenu du fichier

Une seule version pour 2 langages de programmation

- Implementation strictement identique en Java et C++



Contactez dès aujourd'hui Emmanuel Blattes, Ingénieur au Service Technique.  
Il vous aidera à évaluer HydraSDO XML avec vos propres fichiers.

Rogue Wave France  
40, rue des Vignobles - 78400 Chatou - Tel +33 (0)1 30 09 78 78

## Outils

# Sybase lance PowerBuilder 11.5 et prépare la suite

L'éditeur Sybase annonce officiellement PowerBuilder 11.5 dont plusieurs présentations avaient été faites en France. Cette version est dans la continuité de la précédente. Une des nouveautés importantes est l'arrivée d'un pilote natif SQL Server 2008 et Oracle 11g. Cela implique un support étendu des types de données, les données types CLR, les données spatiales, etc. On bénéficie aussi d'une nouvelle syntaxe comme *merge*. La haute disponibilité s'intègre désormais à l'environnement. On trouve ainsi des mécanismes de mirroring. Les pilotes

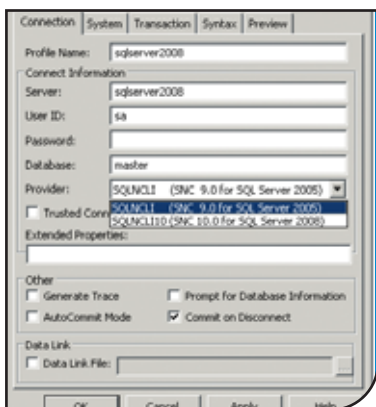
etc. Cette version sera très orientée .net et reverra de fond en comble le modèle de développement, l'éditeur, l'interface, la génération, etc. Le but est aussi de proposer un modèle WPF (celui de .Net depuis la version 3.0). Mais le modèle actuel de développement sera conservé même s'il est clair qu'il tendra à disparaître. La data Window sera elle aussi entièrement recodée en code manager .Net, le .Net CLS est intégré et une interopérabilité entre PowerScript et .Net sera assurée. Côté éditeur, on bénéficiera aussi d'une refonte totale en WPF. Mais comme l'indique l'éditeur, le projet est en pleine maturation et conception.

■ François Tonic

## Sybase veut révolutionner la mobilité

L'éditeur commence à dévoiler son futur environnement de développement pour applications mobiles : SUP pour Sybase Unwired Platform. Il s'agit de produire des applications, une plate-forme indépendante de tout système. On code, on déploie. Il inclut un mécanisme de fonctionnement en connecté / déconnecté, avec synchronisation automatique. Une bêta est prévue courant septembre prochain. L'outil sera utilisable dans Eclipse et Visual Studio.

SUP vise les applications d'entreprise en fournissant différents services, notamment envers les données et les processus métiers. Selon l'éditeur, SUP combine donc un L4G et les bonnes pratiques. Il supporte Windows Mobile, BlackBerry, pour le moment. Il repose sur le modèle MVC et côté middleware, s'appuie sur Java 5 et Tomcat. Le repository est basé sur Sybase SQL Anywhere. On dispose du support de JMX Server, Flex 3 et Flash 9. Côté langage, SUP fournira un méta langage qui doit assurer cette portabilité. On devrait en savoir plus dans quelques semaines.



pilote SQL Server 2008 natif

natifs sont là pour tirer parti au mieux des données de la base de données. Côté développeur, on dispose d'un nouveau data window avec plus de fonctions de reporting, de présentation de données (ex. : tooltips). On bénéficie aussi du Rich Text Edit, permettant d'avoir un texte riche (style, police, taille, etc.) dans un formulaire. On notera aussi une barre d'outils améliorée, le support de DirectX 9, permettant de réaliser des graphes 3D, la transparence. Le format PNG est lui aussi supporté dans la v11.5.

## PowerBuilder 12 : une refonte totale !

Le véritable Big Bang de PowerBuilder aura lieu en 2009 avec la v12 qui marquera une véritable rupture avec l'actuelle lignée. La v12 s'appuiera totalement sur Visual Studio Shell. Il s'agit d'une coquille vide dans laquelle l'éditeur y met ses fonctions, plug-in,

## PowerAMC 15 : bientôt là

Depuis plusieurs mois, les premières bêta de la prochaine version de PowerAMC (PowerDesigner pour le reste du monde), la v15, sont disponibles. Cette nouvelle évolution de l'outil de modélisation dispose d'un éditeur spécifique pour le référentiel, de la possibilité de simplifier l'aspect des modèles, d'une nouveauté dans l'importation depuis Visio, de la modélisation de l'ar-

chitecture d'entreprise, d'un diagramme d'analyse d'impact, et de la modélisation de la logique de données. Actuellement le support de SQL Server 2008 est limité, pas de plug-in pour Visual Studio ou Eclipse 3.4. La bêta 2 est actuellement disponible (fonctionnant jusqu'en novembre prochain), uniquement sur Windows. Voilà une belle perspective !

## agenda \

### SEPTEMBRE

- Le 10 septembre, Paris 7e centre de conférences Microsoft **Winwise Solutions Conference**, pour tout connaître des technologies de développement de Microsoft, disponibles et à venir (15 sessions). <http://wsc08.winwise.fr/Accueil>

- Le 11 Septembre 2008, Paris 17e, Palais des Congrès **Salon emploi informatique Les Jeudis** <http://www.lesjeudis.com/salons-informatiques/>

- Conférence annuelle Zend/Php** Du 15 septembre 2008 au 18 septembre 2008 à Santa Clara (USA) <http://www.zendcon.com>

- Du 17 au 20 Septembre 2008, Paris Expo Porte de Versailles **Apple Expo REMIX 2008**

Le rendez-vous européen incontournable du monde Apple. Nouveautés, démonstrations, ateliers... <http://www.apple-expo.com/>

- Du 24 au 25 Septembre 2008, Paris 14e, 17 bd Jourdan Maison Internationale de Paris **Paris Capitale du Libre** <http://paris-libre.org/>

- MSDN & Technet Tour 2008** Du lundi 29 septembre 2008 au mercredi 05 novembre 2008 - Sud Ouest - Toulouse 29 et 30 septembre - Nord - Lille 13 et 14 octobre - Est - Strasbourg 22 et 23 octobre - Centre Est - Lyon 29 et 30 octobre - Sud Est - Marseille 03 et 04 novembre - Ouest - Nantes 05 novembre <http://technet.microsoft.com/fr-fr/cc184917.aspx>

- Du 30 Septembre 2008 au 02 Octobre 2008, Paris La Défense, CNIT

### Salons Solutions 2008

- ERP** : Le Salon des Progiciels de Gestion Intégrés
- CRM BI** : Le Salon de la Gestion de la Relation Client, de la Gestion des Forces de Vente, du Décisionnel et Business Intelligence
- DEMAT** : Le Salon dédié aux solutions de Dématérialisation et d'Archivage en ligne.
- VIRTUAL FORUM**, le 1er Octobre <http://www.groupe-solutions.fr/>

### OCTOBRE

- Le 2 Octobre, Paris, l'événement de lancement **SQL Server 2008/Windows Server 2008 HyperV**, avec Steve Ballmer. <http://c2.microsoft.fr/f3d7732d6c1549f79da175a7034d0a43/>

- Conférence IBM Rational** le 14 octobre à l'IBM Forum, pour faire le point sur les derniers outils, découvrir Jazz et toutes les nouveautés de la fin de l'année.



**Janus WinForms Controls Suite** de € 498

Ajoutez les interfaces de style Outlook à vos applications WinForms.

- Janus GridEX for .NET (grille de style Outlook)
- Janus Schedule for .NET et Timeline pour .NET (calendrier et journal de style Outlook)
- Janus ButtonBar for .NET et ExplorerBar pour .NET (barres de raccourcis de style Outlook)
- Janus UI Bars & Ribbon Control (menus, barres d'outils, commandes ruban Office 2007)
- Avec style visuel Office 2007 de toutes les commandes

**TX Text Control .NET et .NET Server** de € 413

Composants de traitement de texte pour Visual Studio .NET.

- Ajoutez le traitement de texte professionnel à vos applications
- Zone de texte Windows Forms hors droits d'auteur
- Véritable WYSIWYG : tableaux imbriqués, cadres de texte, en-têtes et pieds de pages, images, puces, listes numérotées, zoom, sauts de section, etc.
- Chargez, enregistrez et modifiez : DOCX, DOC, RTF, HTML, TXT et XML

**DXperience Enterprise** de € 799

Tous les outils DevExpress ASP.NET, WinForms et IDE Productivity en un.

- Abonnement de 12 mois pour tous les produits et mises à jour Developer Express et accès aux versions bêta en développement actif
- Composants et outils : grilles, entrée de données, outils d'écriture de code, analyse de données, graphiques, navigation/disposition, planification, solutions reporting, bibliothèques d'impression, outils de remaniement, bibliothèques ORM

**IBM® Rational Functional Tester** de € 5.242

Test fonctionnel avancé automatisé d'applications.

- Automatisation des tests basés sur les données
- Éditeur de puissance industrielle pour création et personnalisation de tests
- Conçu pour applications Java, Web et VS.NET WinForm
- Technologie et reconnaissance de modèles ScriptAssure pour des scripts de test résistants en cas de changement fréquent d'interface utilisateur

# PHP 5.3 : toutes les nouveautés

A l'heure où le support de PHP 4 se termine c'est PHP 5.3 qui est à nos portes : on peut espérer une version finale pour la fin de l'année. Actuellement les équipes qualité du PHPGroup testent une alpha2 dont la liste des innovations n'a cessé d'augmenter au fil des mois.

Plus les mois passaient, et plus les améliorations prévues de PHP 6 étaient incluses en PHP 5.3, rallongeant la durée de publication. Ce cercle vicieux classique nous conduit à une publication massive, mais attendue longtemps, trop longtemps.

En résumé, beaucoup de travail a été porté sur la modernisation de PHP. Au menu :

- Support des espaces de noms, les namespaces
- Fichiers d'archives PHP : Phar
- Ajout de mysqlnd, fileinfo, intl, sqlite3
- Ajout de `?:`, des goto et du late state binding.
- Corrections dans SOAP, pcntl, spl, openssl, oci8, mbstring et hash
- Profileur de code XSLT
- Fonctions lambda et fermetures
- Retrait des ticks, ze1.compatibility, et des versions avant Windows 2000

## Les espaces de noms

Les espaces de noms sont probablement la fonctionnalité la plus attendue de PHP 5.3, et même, certains diront, sa raison d'être. Les espaces de noms ont pour objectif de guérir PHP de son abcès de noms-de-classes-si-longs-qu'ils-ne-finissent-plus.

Avec l'arrivée d'une programmation-objet solide et moderne, PHP a vu fleurir des frameworks objet, avec de nombreuses classes. Mais PHP ne proposait qu'un seul discriminant de classe, son nom. Pour éviter que deux classes qui pourraient porter le même nom entrent en conflit, il fallait une solution qui les distingue. XML\_Beautiflier\_Renderer une classe PEAR qui est rangée dans le dossier XML/Beautiflier/. Mais comme il pourrait y avoir une autre classe qui porte le nom de *Renderer* dans l'arborescence de l'*include\_path*, il faut trouver un moyen de faire la différence entre les classes à coup sûr. En l'occurrence, il existe *Var\_Dump\_Renderer*, *Text\_Diff\_Renderer* et même *ezcGraphAxisCenteredLabelRenderer* dans les composants eZ.

Les espaces de noms se présentent donc comme un outil pour hiérarchiser les classes, et les ranger proprement. Prenons quelques classes alphabétiques, rangées dans un dossier namespaces, avec le code suivant :

```
<?php

namespace Y;

class A {
    function __toString() {
        $retour = "Classe : " . __CLASS__ . "<br/>";
        $retour .= "Namespace : " . __NAMESPACE__;
        return $retour;
    }
}
```

Tout d'abord, il y a le mot *réservé namespace*, qui indique le paquet dans lequel on range la classe. Dans cet exemple, la classe A est dans l'espace de noms Y. On pourrait aussi utiliser plusieurs niveaux d'espace en intercalant des `::` entre les noms :

```
namespace A::Y;
namespace XML::Utils::Date::Country::Renderer;
```

Le mot clef " namespace " doit être utilisé en tout début de fichier de définition, et une seule fois au plus dans chaque fichier. Vous pouvez définir des espaces de noms distincts pour chaque fichier, mais il est alors unique, et valable pour toutes les définitions de classe et de fonctions du fichier.

Une fois les espaces de noms définis pour chaque classe, on peut alors utiliser les instructions suivantes pour accéder à des classes qui portent le même nom.

```
<?php
include('namespaces/X/A.php');
include('namespaces/Y/A.php');
include('namespaces/A.php');

$a = new A();
print $a;
// On accède à la classe de A.php

$x = new X::A();
print $x;
// On accède à la classe de X/A.php

// importation d'une classe
use Y::A;
$y = new A();
print $y;
// On accède à la classe de Y/A.php par défaut

// pour accéder à la classe A (sans espace), il faut alors préciser ::
$a2 = new ::A();
print $a2;
// On accède à la classe de A.php

// on peut aussi utiliser un alias pour désigner un espace
use X as mon_espace;
$z = new mon_espace::A();
print $z;

?>
```

Avec cette approche, finis les noms à rallonges comme *ezcGraphAxisCenteredLabelRenderer* : on peut les ranger dans



`ezc::Graph::Axis::Centered::Label`. Les espaces de noms fonctionnent aussi avec les fonctions : les fonctions qui sont rangées dans un fichier, sous une instruction namespace. La portabilité du code actuel est assurée : par défaut, toutes les classes sont rangées dans un espace de noms global, accessible avec l'opérateur `::`, sans préciser de noms. Cela permet de retrouver les classes standard et les fonctions natives de PHP, et d'accéder à vos anciennes définitions.

## Phar : comme du jar

Phar est à PHP ce que Jar est à Java : une archive qui contient du code. Prosaiquement, c'est un fichier TAR compressé, qui contient différents fichiers, auxquels PHP peut accéder directement. Pour distribuer une application PHP, il est alors possible de constituer une archive PHar, et de la diffuser. Pour créer une archive, on utilise PHP :

```
<?php

$phar = new Phar(__DIR__.'classes.phar', 0, 'classes.phar' );
$phar->buildFromDirectory('includes', '/\.(gif|php)$/');
$phar->compressFiles( Phar::GZ );
$phar->stopBuffering();
?>
```

Ce petit script produit un fichier d'archives, appelé " classes.phar " dans le dossier courant, à partir du dossier d'inclusions " includes ". À l'utilisation, phar est très souple. Phar accepte aussi les inclusions partielles d'archives, ce qui est pratique quand vous avez besoin de charger une seule classe dans une bibliothèque complète. Phar a été l'objet de beaucoup de soins pour sa conception, et ses fonctionnalités vont sûrement sauter aux yeux des éditeurs d'applications : rassembler les informations, compresser, chiffrer les codes, etc. Pour les performances, on manque encore d'ateliers de performances, mais cela ne va pas manquer de fleurir dans les prochains mois. C'est un code qui a beaucoup mûri, et se révèle bien stable.

## Fonctions lambda et fermetures

Les fonctions lambda sont des fonctions qui servent de manière opportuniste, mais presque pas ailleurs dans le reste du code. C'est notamment une situation que l'on rencontre quand on a besoin de fonction de rappel (les callback), comme avec `preg_replace_callback()`, ou `usort()`. Les fermetures (ou closures, en anglais) sont en fait des fonctions lambda, qui partagent des variables avec le code principal. Lors de leur déclaration, on indique les variables qui seront utilisées dans le cadre de la fermeture, et une copie de celles-ci sera transmise depuis le contexte d'appel. Comme dans l'implémentation d'un trim sur le tableau ci-dessous :

```
<?php

$array = array(' a ', ' b- ');

function array_trim ($array, $chars = " \t\n\r") {
    $map = function ($item) use ($chars) {
        return trim($item, $chars);
    };
    return array_map ($map, $array);
}
```

```
print_r(array_trim($array, ' -'));
?>
```

Le paramètre `$chars` de `array_trim()` est transmis à la fonction lambda via le mot-clé `use`, pour pouvoir faire un appel correct de la fonction `trim()`. Il est transmis par défaut par valeur, ce qui évite à la fonction lambda de polluer les valeurs de la fonction appelante, mais PHP permet aussi de faire un passage par référence, pour modifier la variable simultanément dans les deux contextes.

### Et encore

- La bibliothèque d'accès à " MySQLnd " permettant de tirer parti des fonctionnalités de MySQL 5. Permet d'économiser 15 % de mémoire, de gagner 35 % de vitesse et de bénéficier de `mysqli_fetch_all()`.
- Support des GOTO : on peut désormais nommer des emplacements dans le code, et y renvoyer l'exécution du script.
- La directive `zend.ze1_compatibility_mode` permettant de garder le fonctionnement objet PHP 4.
- Support de l'opérateur court " ? : ", lequel permet de tester l'existence d'une valeur, et, sinon, de lui donner une valeur par défaut : `$x = $_GET['x'] ? : " valeur par défaut " ;`
- Ajout de l'extension *fileinfo*, qui permet de vérifier le contenu des fichiers sans se baser aveuglément sur les en-têtes de fichiers. Cela améliorera la sécurité des sites qui acceptent les téléchargements de fichiers.
- Support direct de SQLite 3 : plus besoin de passer par PDO.
- Internationalisation : une extension, liée à libICU, pour piloter les localisations.
- Amélioration de la gestion de la mémoire, et notamment gestion des références cycliques
- Une nouvelle erreur utilisateur `E_USER_DEPRECATED` peut être lancée avec la fonction `trigger_error()`, et permet d'indiquer clairement que certaines fonctions utilisateurs sont obsolètes.
- Deux fonctions de tables, `array_replace` et `array_replace_recursive` permettent désormais de faire des remplacements de valeurs et clés dans un tableau.
- Le support de l'algorithme Diffie-Hellman a été apporté à l'extension OpenID, pour pouvoir supporter plus facilement OpenID depuis PHP.
- Les versions antérieures à Windows 2000 ne sont plus supportées.
- PHP 5.3 corrige 82 bugs.

PHP 5.3 est un bon avant-goût de ce que sera PHP 6 : une grande partie des améliorations de PHP 6 ont été intégrées dans cette version. Les namespaces et l'extension phar sont autant de fonctionnalités demandées depuis longtemps par la communauté. L'ensemble du PHPGroup fait preuve d'une belle réactivité et continue d'inscrire dans la durée la réussite d'un des projets phares de l'OpenSource : PHP.



■ **Damien Seguy** est manager de Alter Way Consulting, le département d'expertise en logiciels libres du groupe Alter Way. Co fondateur de l'Association française des utilisateurs de PHP. Auteur de trois livres consacrés à PHP et est co-auteur des certifications Zend PHP.



■ **Julien PAULI** est consultant chez Anaska, la marque formation du groupe Alter Way. Son expertise porte principalement sur l'optimisation PHP. Contributeur actif au projet ZendFramework.

# Jazz : la nouvelle plate-forme collaborative est enfin là !

Cet été, le petit monde du collaboratif est en pleine effervescence depuis la sortie officielle de la plate-forme Jazz et des premiers outils de collaboration basés sur cette technologie. L'ambition d'IBM Rational sur ce marché est immense et Jazz risque bien de bouleverser rapidement la situation.

■ François Tonic

Jazz, en développement depuis plusieurs années chez IBM Rational, est une plate-forme de collaboration pour les équipes, particulièrement étudiée pour les équipes géographiquement distantes (sur différents sites, en offshore, Paris – Province, province-province, etc.). L'objectif premier est de fournir une brique technologique commune à tous les outils ayant besoin d'implémenter des fonctions collaboratives. Si on se risquait à faire une comparaison, ce serait l'Eclipse de la collaboration. D'ailleurs, l'éditeur n'a pas caché sa volonté de voir se créer un écosystème identique avec des outils commerciaux et open source, des plug-in, etc.

## Comment fonctionne Jazz ?

Jazz repose sur trois éléments :

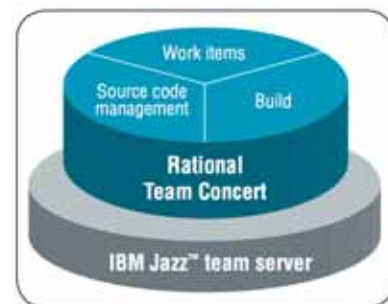
- **Collaboration** : Jazz fournit les relations, les Works items, les flux de collaboration, la prise en charge de la messagerie instantanée (ex. : Jagger), le cycle de vie,
- **Gouvernance** : la plate-forme permet aux équipes de prendre et d'utiliser les méthodes agiles, les bonnes pratiques, selon la granularité souhaitée. On définit son propre process de gouvernance.
- **Productivité** : la plate-forme permet de

suivre au jour le jour l'évolution des tâches attribuées, la réalisation, le retard. Possibilité de gérer plusieurs équipes, plusieurs projets. Modification dynamique des tâches, des équipes, en toute indépendance de la localisation. Jazz a été pensé dès le début pour le web. Il se base sur l'architecture OSGi et ses standards. Le client peut être aussi bien web que stand alone (idem pour l'administration). Jazz est avant tout une interface middleware ouverte que l'on peut implémenter dans ses outils. Car seul, il ne sert pas à grand chose. Il faut une implémentation.

Il utilise les concepts, techniques et standards du Web 2 et des réseaux sociaux. Il s'agit d'une plate-forme orientée Gestion de cycle de vie des applications (ou ALM en anglais). Jazz vise à rendre plus souple l'organisation des projets en équipe, le travail collaboratif en proposant une couche technique unique. Une architecture Jazz est assez simple à comprendre. Il faut disposer d'un serveur Jazz avec un serveur Java EE, Eclipse, le noyau Jazz, une base de données (servant de référentiel, de lieu de stockage), d'une interface web. Tout cela forme le Jazz Team Server. Là-dessus se greffe un outil de messagerie instantané de type Jabber, puis le client Jazz : application desktop ou web. Bref, Jazz reprend le concept d'architecture orientée services (SOA), s'appuyant fortement sur les services, les services web.

## Les composants de Jazz

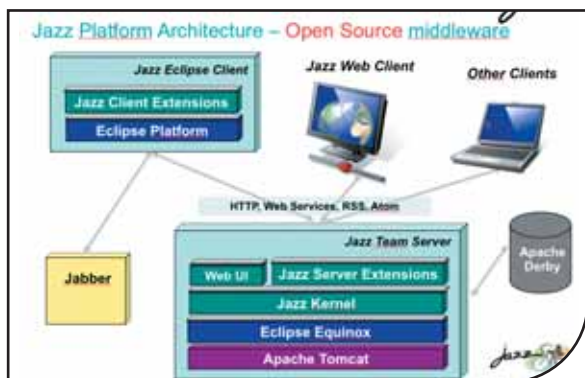
Jazz faisant de l'ALM ne vient pas seul. Il dispose de différents modules. Et rien n'empêche de compléter par ses propres modules. Par défaut, on dispose d'un contrôle de source, d'un Team Build, des



Work Items, de l'état du projet, d'une couche d'interopérabilité, des mécanismes collaboratifs, des processus, d'un référentiel, d'un planning d'itération (lié aux work items). Le but est de pouvoir disposer de plug-in Jazz pour des outils, environnements non IBM Rational. Comme ce que l'on a pour Eclipse, Visual Studio, NetBeans.

## Team Concert : le premier avant la déferlante !

Rational Team Concert 1.0 est le premier outil IBM "Jazz ready". Pour IBM, les outils actuels de collaboration, de développement en équipe sont complexes à déployer, lourds et pas toujours très maniables. Jazz doit changer cela ! Team Concert arrive en trois éditions : Express-C (gratuite), Express et Standard. La première est limitée à 10 utilisateurs, la seconde à 50, la standard en supporte 250. Pour la base de données, seules DB2 et Oracle sont actuellement supportées, Express-C se limite à Derby. Même choix limité pour le serveur d'application (inclus en standard). Ensuite les différences se font sur les fonctions avancées comme le reporting, le tableau de bord, les connecteurs ClearCase / ClearQuest, LDAP, etc. Chaque version arrive avec 3 licences utilisateurs gratuites. Pour les utilisateurs supplémentaires, il faut payer (de 600 euros à plus de 3 900 euros chaque licence selon la version, en plus de la licence serveur, à partir de 6 000 euros). L'outil est très complet dans la gestion des projets, des work items. On peut finement définir les droits d'accès de chacun, on contrôle strictement l'activité de chaque membre de l'équipe, l'avancement des travaux et surtout, il s'interface avec d'autres outils IBM comme Build Forge. Cependant, la maîtrise de l'environnement nécessite une solide formation car l'interface n'est pas toujours optimale et la multitude des options et fonctions rend parfois difficile la compréhension.





Extrême  
Java

Struts2

UML

Hibernate

valtech  
training

Gestion  
de projet

Scrum

LinQ

.Net

## Au plus court vers vos nouvelles compétences

### Architecture et intégration

- Introduction à l'architecture orientée service (SOA) (1 jour)
- Ingénierie logicielle objet (3 jours)
- Introduction aux architectures et technologies du Web (1 jour)
- Architectures .Net multi-niveaux (3 jours)
- Urbanisation du système d'information (2 jours)
- Architecture orientée service (SOA) (3 jours)
- Architecture d'entreprise avec Java EE (4 jours)

### Développement Java et C++

- Introduction technique à Java (1 jour)
- Programmer en utilisant les aspects et les Design Patterns (3 jours)
- Java et la conception objet (5 jours)
- Développement d'un client riche avec SWT et Eclipse RCP (3 jours)
- Atelier Java avancé (5 jours)
- Eclipse, créer son environnement de développement intégré (2 jours)
- Programmation intensive avec Java (5 jours)
- Extrême Java (4 jours)
- Développer une application Corba (4 jours)
- L'essentiel de C++ et la conception objet (5 jours)
- Programmation efficace et avancée en C++ (5 jours)

### Microsoft .Net

- C# et la conception objet (5 jours)
- Programmation avec VB.NET et conception objet (5 jours)
- Programmation intensive avec le Framework .Net (5 jours)
- Développement d'applications Web avec ASP.NET (5 jours)
- Développement d'applications Windows Forms sur la plate-forme .Net (5 jours)
- Développer des applications graphiques avec WPF (3 jours)
- Gestion des données avec LinQ (3 jours)
- Communication inter-applicative avec WCF (3 jours)
- Développer un client riche avec Silverlight 2 (3 jours)

### Analyse, conception et modélisation avec UML

- Introduction technique à l'analyse, la conception et la programmation objet (1 jour)
- Introduction à UML (1 jour)
- Concevoir avec les Design Patterns (5 jours)
- Modélisation métier avec UML (3 jours)
- Analyse et conception avec UML (5 jours)
- Modélisation des systèmes complexes avec UML 2 et SysML (3 jours)
- Modélisation efficace des exigences avec les cas d'utilisation (2 jours)
- Analyse orientée objet avec UML (2 jours)
- Introduction technique à SysML (1 jour)
- Modéliser les besoins et analyser avec UML (4 jours)
- UML avec Enterprise Architect (2 jours)

### Frameworks Java EE

- Concevoir et développer des EJB 2 (5 jours)
- Développer une application Java EE avec les EJB 3 (5 jours)
- Gestion de la persistance avec Hibernate (3 jours)
- Mise en œuvre du Framework Seam (5 jours)
- Développement avec le Framework Spring (3 jours)
- Gestion avancée de la persistance avec Hibernate (2 jours)

### Gestion de projet

- Gérer des projets avec un processus itératif (4 jours)
- Coacher son équipe projet (3 jours)
- Du recueil des besoins aux exigences : rédiger le cahier des charges (2 jours)
- La conduite de projet (3 jours)
- Gestion de projet informatique (3 jours)
- Manager des hommes dans le cadre d'un projet (2 jours)
- Management de projet (5 jours)
- MSProject (3 jours)

### Oracle

- Introduction technique (1 jour)
- Exploitation (4 jours)
- SQL (3 jours)
- PL / SQL (3 jours)
- Optimisation des requêtes (2 jours)
- Administration (5 jours)
- Tuning (3 jours)

### XML et Web Services

- Introduction à la technologie XML (1 jour)
- Introduction aux technologies Web Services (1 jour)
- Développer avec XML (3 jours)
- Développer une application XSL (2 jours)
- Développer des applications Web Services en Java (3 jours)
- Développer des applications XML avec Java (2 jours)

## 90 formations au développement logiciel

chez vous  
ou à Paris, Toulouse, Lyon, Grenoble,  
Genève, Bruxelles, Luxembourg

### Méthodes et pratiques agiles

- Test Driven Requirement ou la gestion des exigences dirigée par les tests (2 jours)
- Test Driven Development ou la programmation pilotée par les tests en Java (2 jours)
- Stratégie de test, vérification et validation (3 jours)
- Méthodes agiles de développement logiciel (1 jour)
- Gérer les projets agiles avec Scrum (2 jours)
- Mettre en œuvre le Lean Software Development (3 jours)
- Tests unitaires en Java avec JUnit (2 jours)
- Devenir Scrum Product Owner (2 jours)
- Qualité du code dans les projets Java (2 jours)
- Gestion de configuration logicielle avec Subversion (1 jour)
- Usine logicielle, des concepts à la pratique (3 jours)

### Développement Web

- Développement de pages Web avec HTML et CSS (2 jours)
- Développement client avec JavaScript et Prototype (3 jours)
- Développement d'applications Web avec PHP (5 jours)
- Ajax, pour dynamiser le poste client (2 jours)
- Programmation Ajax avec Dojo Toolkit (2 jours)
- Développement d'applications Web avec Struts 2 (3 jours)
- Hacking des applications Web (1 jour)
- Développer des applications avec Adobe Flex 3 (5 jours)
- Conception d'applications Web d'entreprise avec Java EE (5 jours)
- Développement d'applications Web avec JSF (3 jours)
- Développement de composants JSF (2 jours)
- Utilisation du Framework Struts pour le développement d'applications Web (3 jours)
- Développer une application Web avec Ajax et GWT (3 jours)
- Développer des applications pour Adobe Integrated Runtime (AIR) (2 jours)



www.valtech-training.fr  
+33 (0)1 41 88 23 00 - +33 (0)5 62 47 52 00  
info@valtech-training.fr

# Tout savoir sur SQL Server 2008

La nouvelle version de SQL Server, la base de données de Microsoft, est disponible depuis quelques semaines. Dans cet article nous allons passer en revue une grande partie de ses nouvelles fonctionnalités.

Cette version 2008 ne bouleverse pas les avancées apportées par la version 2005 mais arrive avec une myriade de nouvelles fonctionnalités. Microsoft souhaite d'ailleurs adopter une approche orientée "features" pour faire évoluer ses produits et faciliter les mises à jour. Ces nouveautés ont pour vocation d'améliorer la base de données de Microsoft sur les axes de la fiabilité, pour accroître la capacité à faire tourner des applications critiques d'entreprise ; la productivité, pour offrir aux développeurs et aux administrateurs plus de simplicité, pour en faire plus en moins de temps ; l'intelligence, pour permettre à chaque utilisateur d'exploiter l'information la plus pertinente.



## Nouveautés du langage

Commençons par les "petites" nouveautés, celles du langage T-SQL (le SQL maison). Ce ne sont pas les plus grosses ni les plus attendues mais elles améliorent l'utilisation au quotidien et cela va réduire sensiblement la taille de nos scripts :

- **Déclaration et assignation de variable** : on peut maintenant en une seule instruction déclarer et attribuer une valeur à une variable.

```
DECLARE @i AS int = 123
```

- **Opérateurs** : on peut maintenant faire une opération et réaffecter le résultat en une seule instruction

```
SET @i += 1
```

- **Instruction MERGE** : cette nouvelle instruction permet de fusionner 2 jeux de données. Très concrètement, elle permet par exemple de traiter l'insertion et la mise à jour en une seule fois.

```
MERGE dbo.Contact AS Destination
USING (SELECT * FROM dbo.Donnees_a_Inserer) AS Source
ON (Destination.ContactId = Source.ContactId)
WHEN MATCHED THEN
    UPDATE SET Destination.Adresse = Source.Adresse, Destination.Date = GetDate()
WHEN NOT MATCHED BY TARGET THEN
    INSERT (ContactId, Adresse, Nom)
VALUES (Source.ContactId, Source.Adresse, Source.Nom) ;
```

- **Insertion multiple** : l'instruction INSERT s'enrichit avec la possibilité d'ajouter plusieurs enregistrements en même temps

```
INSERT MaTable(Nom, Prenom)
VALUES ('Riehl', 'Jean-Pierre')
VALUES ('Pertus', 'Sébastien') ;
```

## Du nouveau pour les données

### Nouveaux types

- **HierarchyID** : ce type permet de modéliser plus facilement des structures arborescentes comme le très célèbre Employé-Manager. Il offre aussi de nombreuses méthodes pour manipuler les données efficacement (recherche d'un ancêtre, détermination du niveau, etc.)

- **Date & Time** : il y a 4 nouveaux types liés aux données temporelles qui voient le jour. Les plus simples sont le type Date qui ne stocke que la date du jour mais pas l'heure, et le type Time qui lui stocke uniquement l'heure avec une précision de 100 nanosecondes. Apparaissent aussi 2 nouveaux types DateTime. Le premier, datetime2, qui a une précision beaucoup plus importante (100ns) et le second qui prend en compte les fuseaux horaires.

- **Geography et Geometry** : attendu depuis longtemps, SQL Server s'enrichit du support des données spatiales. Ces 2 types permettent respectivement de gérer des coordonnées géographiques, c'est-à-dire relatives à la Terre, et des données géométriques, à



savoir un point sur un plan. Ces deux types sont livrés avec de nombreuses méthodes de manipulation et un système d'indexation.

- **Table-Valued Parameter** : il existait déjà un type Table que l'on pouvait manipuler mais on ne pouvait pas le passer comme paramètre d'une procédure stockée ou d'une fonction. C'est possible dans cette nouvelle version avec les TVP.

```
CREATE TYPE UserIds AS TABLE (
    UserId VARCHAR(50));
CREATE PROCEDURE spDeleteUsers
    @UsersToDelete UserIds READONLY
AS
[...]
```

## Le " spatial " : la grosse nouveauté

Les fonctionnalités spatiales étaient très attendues dans une base de données de la taille de SQL Server. Pour cette nouvelle fonctionnalité, les équipes de Microsoft s'appuient sur les standards de l'Open Geospatial Consortium et fournissent une solution complète, clef en main et intégrée. Les types spatiaux sont facilement manipulables et rendent presque naturel l'ajout d'informations géographiques ou vectorielles dans une application. La requête ci-dessous montre comment retrouver des points dans une zone donnée.

```
Declare @zone geography = geography::STPolyFromText('POLYGON
((43 3.2, 43 3.5, 43.2 3.5, 43.2 3.2, 43 3.2))', 4326)

Select *
From CustomerPlaces
Where @zone.STIntersects(CustomerPlaces.Localization) = 1
```

L'intégration avec ADO.NET et les applications .NET est tout aussi triviale puisque les mêmes types geography et geometry (ce sont des types CLR) existent dans le framework :

```
SqlGeography g = SqlGeography.Point(Latitude, Longitude, 4326);
```

Pour un rendu plus visuel, on pourra s'appuyer sur des technologies comme Virtual Earth ou MapPoint.

Ces nouveaux types ouvrent la voie à des applications localisées, voire à un système d'information localisé.

## Nouvelles façons de stocker les données

- **FILESTREAM** : on peut stocker depuis longtemps des fichiers ou de larges objets binaires. Depuis 2005, on stocke ces BLOB dans un champ de type varbinary(max). Les options de stockage des tables permettent de gérer des gros volumes de données sans difficulté, avec un bon niveau de performance. Cependant, il reste un point sur lequel SQL Server n'est pas très à l'aise, c'est une lecture de BLOB

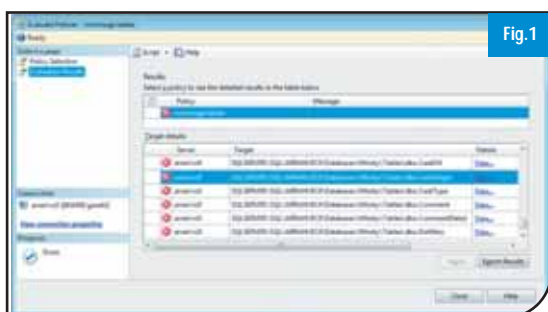


Fig.1

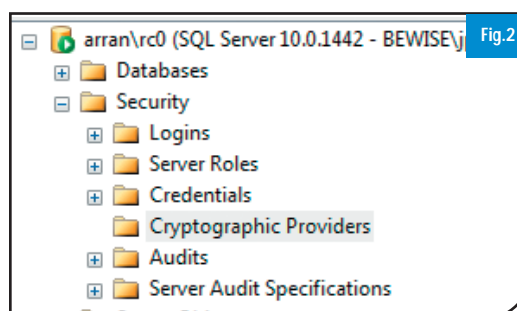


Fig.2

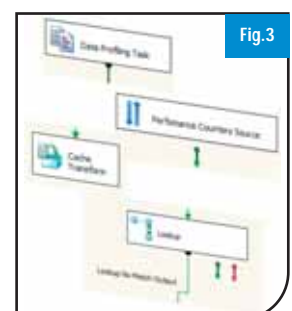


Fig.3

## " Le plus gros lancement serveur de l'histoire de Microsoft "

C'est en ces termes que Lionel Billon (chef de produit SQL Server Microsoft France) parle de la version 2008 de la base de données maison.

**Programmez ! : Quels sont les éléments à retenir de cette nouvelle version 2008 de SQL Server ?**

**Lionel Billon** : Tout d'abord, il s'agit d'une version de continuité avec l'édition 2005, puis il y a les performances et la partie Business Intelligence et relationnelle. Nous apportons beaucoup de nouveautés, comme les données géospatiales, le file Stream, l'administration, la possibilité pour l'utilisateur de créer son rapport de BI, les mécanismes de compression.

**Programmez ! : Quelle ambition avez-vous pour cette version ?**

**Lionel Billon** : Nous montrons que notre plate-forme est idéale pour les applications critiques, le datacenter. L'ensemble SQL Server + Windows Server + Visual Studio permet de générer très rapidement des développements de nouvelle génération, notamment pour le web.

**Programmez ! : La BI, le décisionnel en général, est un axe important pour Microsoft, est-ce toujours le cas avec cette 2008 ?**

**Lionel Billon** : Oui. Nous proposons une solution complète avec de l'ETL, de l'OLAP, etc. On peut créer soi-même ses rapports. SQL Server 2008 avec Office constitue l'offre de business intelligence la plus complète du marché.

en mode stream. En effet, dans le cas de très gros fichiers (de la vidéo par exemple) SQL Server monopolise beaucoup de ressources car il doit placer tout le champ dans ses buffers de transfert. FILESTREAM permet de stocker ce type de données sous la forme de fichiers sur un système NTFS classique. Ce mode de stockage permet de pouvoir récupérer un pointeur sur le système de fichier et de lire les données de façon streamée, comme un fichier classique :

```
CREATE TABLE dbo.VideosData (
    Id INT PRIMARY KEY,
    Mpeg4Data VARBINARY(MAX) FILESTREAM NULL)
```

- **SPARSE columns** : le concept de Sparse columns représente les colonnes à faible densité, par exemple les attributs d'un contact dans une base de CRM qui peuvent être très nombreux et pourtant très peu renseignés. Le fait d'indiquer à SQL Server que ces attributs doivent être stockés dans des colonnes de type SPARSE permet de gagner l'espace normalement occupé par les valeurs nulles. Ce gain n'est pas tant pour le stockage que pour la manipulation des données qui s'en trouve améliorée (il y a moins d'octets à lire).

```
CREATE TABLE dbo.ClientCRM (
    ClientID int NOT NULL IDENTITY (1, 1),
```

```
Nom varchar(50) NOT NULL,  
Poids int SPARSE NULL,  
Yeux varchar(50) SPARSE NULL,  
Taille numeric(3, 2) SPARSE NULL)
```

Les sparses columns s'utilisent aussi avec la notion de Columns Set qui permet de récupérer ou d'insérer plus facilement des jeux de colonnes disposant d'une valeur sans avoir à les manipuler toutes explicitement.

## Nouvelles façons de requêter des données

- **iFTS** : iFTS est l'abréviation de Integrated Full-Text Search. La recherche Full-Text existe depuis un moment dans SQL Server mais se base sur une architecture inter-processus. En effet, la recherche s'appuyait sur le service Windows Search, qui réside en dehors du giron de SQL Server. Avec la version 2008, la recherche a été intégrée dans SQL Server. Cette intégration permet des gains de performances énormes car l'optimiseur peut mixer les opérations de recherche full-text avec des opérateurs de jointures ou de filtrage classique dans le plan d'exécution. Cette nouvelle version se voit aussi enrichie de petites fonctionnalités qui facilitent la gestion d'une solution de recherche comme la gestion plus simple des mots interdits ou la consultation des mots de l'index.

- **Index filtrés** : Il est désormais possible de poser des index sur une partie seulement des données d'une table. On rajoute juste une clause WHERE dans la création de l'index et celui-ci ne s'appliquera qu'à un sous-ensemble des données. Ici on n'indexe que les clients qui ont un numéro de téléphone renseigné.

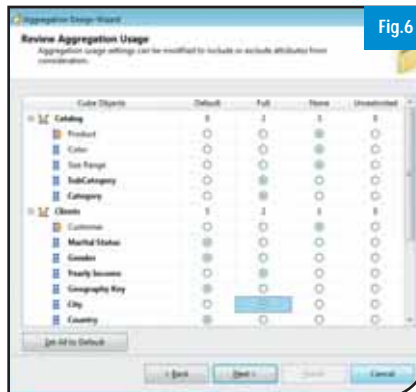
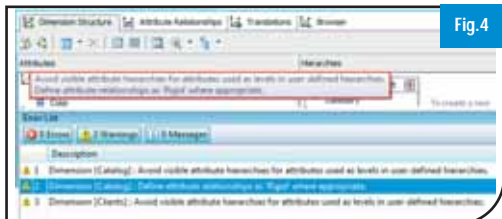
```
CREATE INDEX IX Phonenum ON dbo.Client  
(Phonenum ASC)  
WHERE Phonenum is not null
```

## Nouveautés pour les administrateurs

- **Compression des backup** : cela peut paraître trivial de compresser un backup de base de données mais cette opération devait être faite a posteriori par l'opérateur de sauvegarde. Maintenant, c'est SQL Server lui-même qui peut compresser la sauvegarde au fil de l'eau. Certes, plus de CPU est consommée mais cela permet de manipuler localement de gros backup.

- **Fonctions d'audit** : cette fonctionnalité est similaire à la TRACE mais avec une vocation plus administrative. Elle permet d'auditer certaines opérations sur le serveur et de les loguer dans un fichier ou dans le journal d'événements :

```
CREATE SERVER AUDIT SuppressionAudit TO APPLICATION LOG  
  
CREATE DATABASE AUDIT SPECIFICATION Audit Suppression Produits  
FOR SERVER AUDIT SuppressionAudit  
ADD (DELETE ON dbo.Product)
```



- **Administration par règles** : plus connue sous le nom de Policy Framework, cette fonctionnalité permet enfin aux administrateurs de base de données d'édicter des règles de gestion ou de nommage pour leurs bases et de les faire respecter. On écrit d'abord une condition, par exemple que toutes les tables doivent appartenir à un schéma autre que dbo. On définit ensuite une règle de gestion de cette condition. Enfin, on peut soit l'évaluer sur demande pour avoir un état des lieux. Mais on peut aussi forcer la validité de la règle à tout moment, et ainsi interdire l'exécution d'un script ne la respectant pas. [Fig.1]

- **Data collector** : ce nouveau composant permet de récupérer et de stocker des informations précieuses sur le fonctionnement de SQL Server comme l'utilisation des disques ou de la mémoire. La nouveauté n'est pas dans les compteurs remontés, car ils sont déjà auditables depuis le moniteur de performance de Windows ou depuis les vues systèmes, mais dans le mode de collecte, complètement automatisé et dans la consolidation des données puisqu'un DataWarehouse dédié sert de stockage.

## Nouveautés pour la sécurité

**Chiffrement transparent** : il existait déjà des API de chiffrement et toute une hiérarchie de clés et certificats intégrés à la version 2005 pour gérer la sécurité des données. Cependant, il fallait manipuler ces API en T-SQL au moment de l'utilisation des données. Avec la version 2008, il est maintenant possible de chiffrer toute une base de données de façon transparente pour les utilisateurs.

```
ALTER DATABASE SalairesDB  
SET ENCRYPTION ON
```

**Utilisation de clés externes** : comme indiqué ci-dessus, SQL Server possède toute une hiérarchie de sécurité pour gérer les clés et les certificats. Mais dans un système d'information, on peut disposer d'une gestion centrale des clés de chiffrement (le PKI) et vouloir en tirer parti dans la base de données. SQL Server 2008 peut utiliser un gestionnaire de clés externe pour gérer les clés qu'il utilise. [Fig.2]

## Nouveautés pour les performances

- **Compression des données** : SQL Server 2008 supporte la compression des données à tous les niveaux. Nous l'avons vu avec les backup, mais il est aussi possible de compresser directement les données d'une table, d'une partition ou d'un index.

On a du mal à classer cette fonctionnalité dans la catégorie performance car compression est synonyme de consommation processeur. On attend plus souvent de ce genre de fonctionnalité de gagner de l'espace disque. On en gagne certes mais l'avantage principal apporté par la compression est dans l'exécution des requêtes car on doit lire moins de données sur le disque, ce qui est un des



# Infragistics NetAdvantage®

Un potentiel sans limite pour Windows Forms, ASP.NET et WPF

## Les meilleurs composants pour interfaces utilisateur disponibles

Tous les composants essentiels pour la conception et le style d'applications corporatives irrésistibles

## Une expérience utilisateur hors du commun

Concevez des IHM exceptionnelles et ergonomiques pour multi-plateformes

## Un cycle de développement accéléré

Réduisez vos délais de développement et assurez une livraison rapide avec des composants prêts à l'emploi

## Une utilisation facilitée

Concevez des IHM reproduisant fidèlement le look & feel des dernières technologies Microsoft® avec un minimum de code

## Retour sur investissement

Focalisez vos attentions sur l'aspect métier et permettez à la boîte à outils Netadvantage d'assurer la cohérence, le style et l'utilisation simplifiée de l'interface de vos applications pour Windows Forms, WPF et WEB 2.0

## Support complet pour l'IDE de Visual Studio®

Conçu pour compatibilité avec les plateformes de développement de Microsoft y compris Visual Studio 2008

## Pour de plus amples informations:

[infragistics.com](http://infragistics.com)

 0800 667 307

**Infragistics®**  
Powering The Presentation Layer  
[sales-europe@infragistics.com](mailto:sales-europe@infragistics.com)

grids   scheduling   charting   toolbars   navigation   menus   listbars   trees   tabs   explorer bars   editors   & more



principaux nerfs de la guerre de la performance. A utiliser néanmoins avec discernement...

```
CREATE TABLE Contact
(id int, fullname nvarchar(50) )
WITH (DATA_COMPRESSION = PAGE);
```

Resource Governor : cette fonctionnalité est une des plus attendues dans SQL Server. Le Resource Governor permet de gérer finement l'utilisation du serveur (CPU, mémoire). Fini les requêtes qui monopolisent toutes les ressources. On peut assigner des limitations et des plans de charge aux différentes connexions ou applications. Ainsi, on peut faire en sorte qu'une application critique dispose d'un maximum de puissance de traitement et que des petites applications soient moins prioritaires.

```
CREATE FUNCTION fnPriorisation () RETURNS sysname
WITH SCHEMABINDING
AS
BEGIN
    DECLARE @group AS sysname
    IF (SUSER_NAME() = 'Directeur ')
        SET @Group = 'PRIORITAIRE'
    ELSE IF (APP_NAME() LIKE '%REPORT SERVER%')
        SET @Group = 'Reporting Services'
    ELSE
        SET @Group = 'NON PRIORITAIRE'
    RETURN @Group
END
GO
ALTER RESOURCE GOVERNOR WITH (CLASSIFIER_FUNCTION=fnPriorisation);
```

• **Et aussi...** : on peut citer de nombreuses autres améliorations comme l'optimisation et la meilleure gestion des plans d'exécution, ajouts de Hints pour mieux maîtriser les requêtes, améliorations du parallélisme sur les partitions, l'ajout de CPU à chaud, l'amélioration du mirroring, etc.

## Nouveautés décisionnelles

### Nouveautés de Integration Services

- **Change Data Capture** : cette fonctionnalité sert à optimiser le chargement de DataWarehouse. Elle récupère toutes les modifications faites sur une table et les stocke de façon à pouvoir être exploitées sans impact sur les tables de production.
- **Data Profiling** : un nouveau composant voit le jour pour profiler vos données et récupérer des informations précieuses pour mesurer la qualité de vos données. On peut ainsi récupérer le ratio de valeurs nulles, la distribution des valeurs, etc.
- **Gestion du cache** : on peut maintenant dans un package SSIS utiliser un cache de données quand on doit manipuler les mêmes données à différents endroits. De nouveaux composants permettent de le créer et de l'utiliser dans des flux de données.
- **Composant Lookup** : le composant Lookup est quelque peu amélioré grâce à la gestion du cache. En effet, on peut maintenant maîtriser finement le chargement de la table de référence utilisée pour l'opération de lookup.
- **Compteurs de performance** : une nouvelle source de données apparaît pour récupérer les données des compteurs de performance de Windows. [Fig.3]

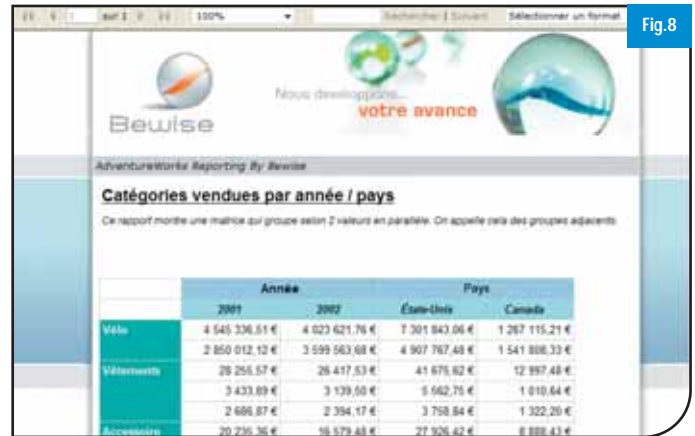


Fig.8

- **Scripting en C#** : les composants de script supportent maintenant le langage C# et s'appuient sur la brique VSTA (Visual Studio Tools for Application) pour supporter la totalité du Framework .NET.
- **Et aussi...** : Microsoft profite de cette nouvelle version pour apporter une amélioration des performances générales notamment grâce à une optimisation du multithreading.

### Nouveautés de Analysis Services

- **Améliorations des performances globales** : dans cette version 2008 de SSAS, Microsoft a cherché à beaucoup optimiser les performances de son moteur multidimensionnel. L'objectif est ambitieux : supporter des DataWarehouse de 100To de données en lieu et place des 25 supportés avec la version 2005. De nombreuses optimisations dans le calcul des agrégats et la gestion des cubes abondent en ce sens.
- **Notion de bonnes pratiques** : une nouvelle fonctionnalité aussi appelée " alertes AMO " apparaît dans Business Intelligence Studio. De la même façon que le correcteur orthographique de Word, elles soulignent les éléments qui ne respectent pas les bonnes pratiques de modélisation et méritent réflexion. Ces avertissements sont débrayables ponctuellement ou définitivement. [Fig.4]
- **Concepteur de dimensions** : le concepteur de dimensions est enrichi d'une représentation graphique des relations entre attributs. Cela apporte une aide visuelle pour la conception de vos dimensions. [Fig.5]
- **Concepteur d'agrégations** : ce concepteur est grandement amélioré puisqu'il permet maintenant de donner des directives précises au moteur OLAP quant à la granularité des agrégations qu'il fait. On peut ainsi décider par exemple de ne pas agréger au niveau jour de la dimension temps car ce niveau est très peu utilisé mais préférer l'attribut couleur des produits. [Fig.6]

**Et aussi...** : d'autres nouveautés sont disponibles dans cette version 2008 d'Analysis Services comme le writeback (écriture dans un cube) dans une partition de type MOLAP ou encore l'amélioration de la pertinence de certains des algorithmes de Data Mining.

### Nouveautés de Reporting Services

C'est sans aucun doute dans Reporting Services que les nouveautés sont les plus visibles, découvrons les plus en détail.

- **Nouvelle architecture** : pas de panique, l'architecture actuelle n'est pas remise en cause. Néanmoins, on a le choix avec cette version 2008 d'installer Reporting Services sans avoir de serveur IIS sur la machine. Ce pré-requis, souvent en contradiction avec les

bonnes pratiques de Microsoft, n'est plus obligatoire. La nouvelle architecture permet à SSRS d'écouter lui-même les connexions HTTP en se branchant directement sur le driver HTTP.SYS de Windows. Cela lui permet de mieux prendre en charge la gestion de la mémoire et les longs rapports.

- **Nouvel outil de création de rapport** : un des défauts dans la version 2005 était les outils de création de rapports. Visual Studio était trop "développeur" et Report Builder n'a pas convaincu du fait de son manque de richesse fonctionnelle. Microsoft s'est fixé comme objectif d'avoir un outil à destination de l'utilisateur final iso-fonctionnel avec Visual Studio, et bien évidemment, le tout avec une ergonomie nettement améliorée.

Il semblerait que le pari soit gagné car Report Designer, qui a le look de la gamme Office 2007, est quasiment le même outil que Visual Studio. Et pour les deux outils, après quelques minutes d'adaptation, on ne peut plus se passer de leur ergonomie intuitive et efficace. Fini les soucis de ne pas savoir à quel niveau d'agrégation on est, quel filtre est mis, quel champ est affiché, etc.

- **Export vers Word** : il est maintenant possible d'exporter un rapport sous le format Word

- **Nouveaux composants graphiques** : Reporting Services 2005 ne brillait pas par la beauté de ses graphiques, très vieillots avec leur look Office 97. Microsoft a acquis la technologie Dundas, célèbre éditeur de composants pour SSRS. Il intègre donc nativement dans sa version 2008 les composants *graphs et jauges* de cet éditeur. Non seulement on peut maintenant faire de jolis graphiques mais la palette fonctionnelle s'enrichit énormément : graphe à double échelle, nouveaux types, échelle discontinue, etc. [Fig.7]

- **Tablix** : avec 2005, nous avions le tableau et la matrice, le premier s'étendant de haut en bas, le second de gauche à droite. Avec 2008, il n'y a plus qu'un composant hybride, le tablix. L'idée est de fusionner le meilleur de la table et de la matrice tout en gommant les limitations des 2 composants. Ce nouveau composant corrige en

effet une grande partie des problèmes de 2005 qui nous contraignaient à faire de la gymnastique dans nos rapports comme les groupes adjacents, les lignes statiques, les totaux, etc. [Fig.8]

## Nouveautés de Management Studio

- **Plug-in** : Management Studio 2005 contenait un moteur de plug-in non documenté et jamais dévoilé. La version 2008 le révèle au grand public. En espérant que la communauté va en faire bon usage et enrichir l'outil de nombreux Power Toys.

- **IntelliSense** : longtemps attendue, c'est une des nouveautés les plus agréables car elle nous sert au quotidien. Une fois goûtée, il est difficile de revenir à 2005 tellement la recherche de tables, de champs, de mots-clés est précieuse.

- **Gestion de groupes de serveurs** : on peut depuis Management Studio travailler sur des groupes de serveurs et passer des scripts en parallèle sur les différents serveurs.

## Conclusion

Sans remettre en cause les acquis de la version 2005, le millésime 2008 apporte tout un wagon de nouveautés et d'améliorations qui sont loin d'être négligeables. Cet article vous en a présenté une grande partie. Nous n'avons pas abordé ici SQL CE 3.5, les frameworks de mapping O/R que sont Entity Framework et Linq to SQL, des modules comme le Change Tracking, les améliorations du XML, etc. Nous y reviendrons dans les prochains numéros et dans Programmez ! Hors Série .Net.

Plus que jamais SQL Server mérite sa place dans le cercle fermé des grandes bases de données d'entreprise, capables de gérer de la petite application au DataWarehouse de 100 To en passant par des applications critiques ou déconnectées.

■ Jean-Pierre RIEHL, Bewise  
MVP SQL Server

# SQL Server 2008 : une offre étudiée

Comme à chaque fois, Microsoft segmente son offre et SQL Server 2008 ne déroge pas à la règle. Six éditions sont disponibles :

- **Standard** : version complète pour la gestion des données. Elle est limitée dans plusieurs domaines comme le nombre des instances, les snapshots de données, le mirroring des données, la compression, l'encryption, ou encore sur la BI.
- **Entreprise** : inclut toutes les fonctions avancées (gestion entreprise, data warehouse, BI, haute disponibilité, etc.).
- **Express** : version totalement fonctionnelle mais limitée dans le nombre de processeurs, la mémoire vive, la taille des bases, etc. Idéal pour les petits projets, ou démarrer en SQL Server.

A cela se rajoutent 3 éditions spécifiques :

- **workgroup** : pour faciliter la gestion et le reporting, pour gérer des branches de

stockages, des branches de rapport, faire de la synchronisation distante.

- **Web** : cette édition mise sur la haute disponibilité, un coût de possession réduit et l'intégration avec LINQ et .Net.
  - **Compact** : édition mobile et embarqué du SGBD. Version gratuite tournant sur PocketPC, Smartphone, XP, Vista.
- Côté prix, Microsoft a depuis longtemps indiqué que l'édition 2008 serait au même prix que la 2005.

## Et le futur ?

Le futur lointain est désormais la prochaine version majeure connue sous le nom de SQL Server vNext qui devrait sortir courant 2011 dont les premières mentions publiques furent faites il y a un an lors des TechEd 2007. Pour le moment rien n'est connu nous en saurons plus dans quelques

## AGENDA À RETENIR !

Une conférence exceptionnelle aura lieu le 2 octobre à Paris avec la présence de Steve Ballmer ! Il y fera la promotion et la présentation de Windows Server, Hyper-V, Softgrid, Visual Studio 2008 et SQL Server 2008. Réservez votre date dès maintenant !

semaines. Plus proche de nous, la disponibilité courant 2009 de SQL Server Data Services faisant partie de l'offre Microsoft Online. Il s'agit tout bonnement de transposer SQL Server en ligne, dans le cloud computing. Sans installer la moindre version localement, sur son serveur, on aura accès

à SSDS directement en ligne ! Selon ses besoins, on pourra ajuster le stockage nécessaire, administrer directement en ligne sa base. C'est une rupture totale avec le modèle traditionnel du SGBD. Il s'agit pour Microsoft d'offrir le même niveau de fonctionnalité que sur la version " normale ". Tout naturellement, SSDS trouvera sa place dans la future plateforme de services de l'éditeur : OSLO, qui sera sans doute disponible l'an prochain. ■



## Gagnez une licence complète SQL Server 2008 !

Microsoft offre aux lecteurs de Programmez et visiteurs du site [programmez.com](http://programmez.com), 300 licences " SQL Server 2008 Edition Standard " (version de production avec 1 accès client, interdite à la revente. L'application aurait une valeur équivalente à 1200 euros environ. Rendez-vous sur [programmez.com](http://programmez.com) !

## Index Multimédia : migrer oui, mais en évitant trop de coupures de services !

Index Multimédia fait tourner plus de 600 sites web et totalise plus de 110 millions de requêtes par jour ! La base de données est donc un chaînon vital pour la société. Christophe Laporte (DBA d'Index Multimédia) nous parle de son expérience sur la nouvelle base de données de Microsoft.

" Nous avons pris du retard dans notre projet de migration. Il est actuellement en cours de réalisation. Mais à chaque nouvelle version nous effectuons la mise à jour (assez) rapidement grâce au programme TAP de l'éditeur, ou quand elle est en RTM. Mais s'il n'y a pas de nouveautés, il n'y a aucun raison de faire la migration. Car cela génère des coupures de données, or, nous sommes dans un modèle de fonctionnement 24/7 ", précise d'emblée Christophe Laporte.

Pour SQL Server 2008, la société n'a pas pris de programme TAP mais les bêtas disponibles. Les tests furent réalisés sur des machines physiques ou en virtualisation. " Nous utilisons Hyper-V sous Windows Server 2008. On peut déployer un serveur en moins de 15 minutes ! ", poursuit le DBA maison.

Pour Christophe, la partie administration fut un enjeu non négligeable. " Pas mal de choses changent. On dispose du Ressource Governor, on peut créer des pools de ressources, répartir la charge. Je note aussi l'auditing pour voir ce qui se passe (exactement) sur un serveur " pointe Christophe. " D'autre part, avec les nouveaux mécanismes de compression, sur le backup (la société possède une base de 2 To, ndr), nous gagnons sur le temps de processing. Notre benchmark indique un vrai confort à l'utilisation ", analyse M. Laporte.

" Nous possédons un très gros existant avec pas moins de 13 instances de SQL Server. Avec la version 2008, nous n'aurons plus que 6 instances ! ", s'exclame Christophe. Et cela va permettre de réduire le coût des licences grâce à une diminution du nombre de processeurs utilisés. Bien entendu, toutes les fonctions de captures de données, de reporting, de BI sont importantes pour une société comme Index Multimédia.

Propos recueillis par François Tonic

## Bewise aime SQL Server 2008



Bewise, spécialiste des technologies Microsoft, est une des sociétés de services en informatique les plus reconnues sur le marché. SQL Server 2008 n'a pas

échappé aux griffes des consultants, développeurs internes mais aussi des clients. Jean-Pierre Riehl, responsable métier décisionnel et gestion des données, a répondu à nos questions.

" Il fallait voir ce qui aller changer, après une version 2005 très réussie. Avant, nous étions sur un cycle de développement de 5 ans, aujourd'hui, c'est 3 ans ", pose d'emblée Jean-Pierre Riehl. Car il ne faut pas changer de version juste pour le plaisir !

Mais toutes les craintes furent rapidement écartées. " Nous avons eu une agréable surprise. Nous avons pu installer les CTP à côté de SQL Server 2005 sans effets de bord ni

de problèmes. Et la migration entre les deux versions se révèle simple. On note aussi que cette nouvelle édition comble plusieurs lacunes de la 2005 ", commente M. Riehl.

" On a (presque) l'impression que cette 2008 est un Service Pack de la 2005 ! Rien n'a été cassé, nous ne sommes pas perdus ". C'était un des craintes, car le passage de 2000 à 2005 fut plus délicat. Mais ce n'est pas pour autant que la société va conseiller et installer partout SQL Server 2008. " Il faut du temps, former les équipes " poursuit le responsable. L'appréciation des nouveautés dépend de leur rôle, de leur fonction.

" L'appréciation est différente selon que l'on est développeur, DSI, DBA. Pour le développeur, une des nouveautés importantes est le type de données spatiales. Nous avons déjà plusieurs projets dessus ! Aujourd'hui, nous trouvons normal de l'utiliser alors qu'il y a encore un an, nous n'en disposions pas ! " S'enthousiasme Jean-Pierre Riehl. " Sur la

partie administration, le Resource Governor était particulièrement attendu. Nous voyons beaucoup de clients avec un unique serveur et si on réalise du reporting (dessous), il mobilise beaucoup de ressources ". La gestion plus fine des ressources est une nécessité pour mieux contrôler les flux et optimiser les performances.

" Si je prends la casquette DBA, nous avons désormais la main sur beaucoup plus d'éléments, on maîtrise ainsi la production. Et c'est bien que Microsoft ait misé sur la fiabilité en notion de 24/7 " précise M. Riehl.

Mais comme rien n'est jamais parfait, il y a forcément des faiblesses, des manques.

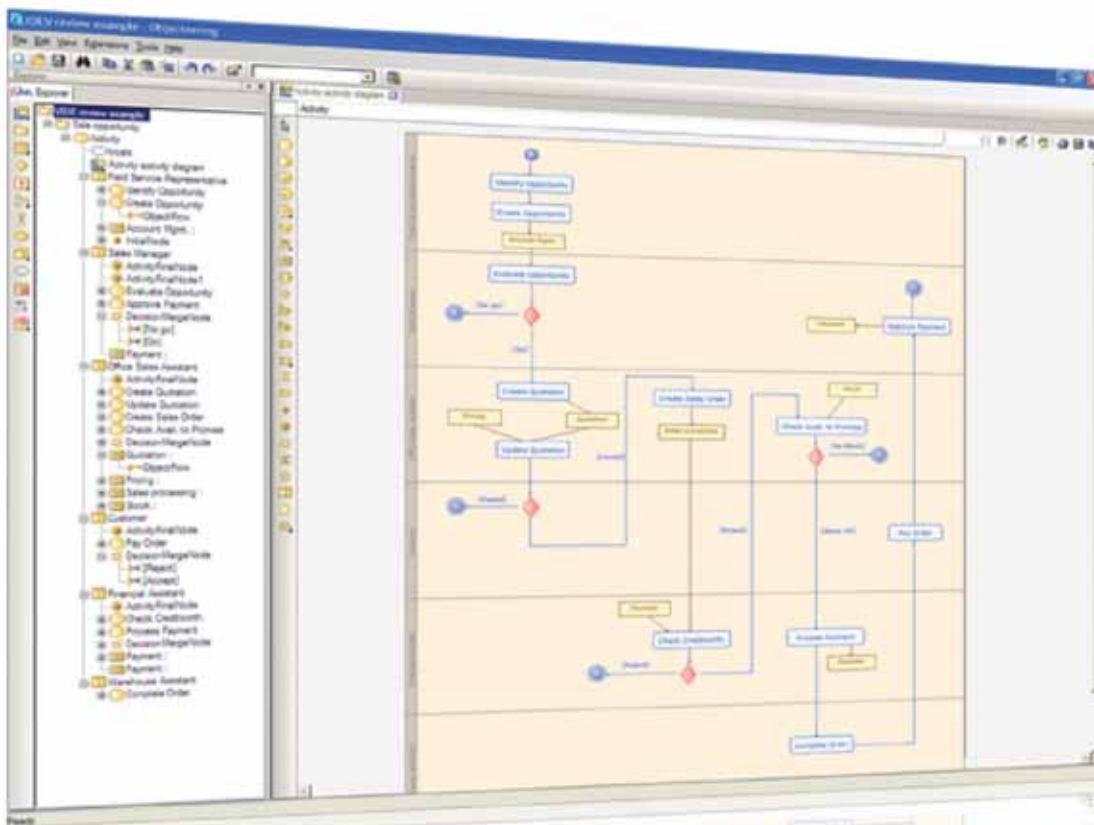
" Peut-être que la fonction des backup aurait pu avoir un outil intégré " nuance notre responsable. Et sur le raccourcissement du cycle de développement, est-ce un problème ?

" Personnellement, je dirais non, même si nous voyons des clients commencer tout juste à migrer sur la 2005. Mais dans la pratique, on peut passer directement de 2000 à 2008 ! ", conclut Jean-Pierre Riehl.

■ Propos recueillis par François Tonic



## Objecteering Modeler



guide vos développements  
par les modèles sur toute  
la portée du système

La technologie MDA associée à une ouverture de l'outillage en Java permet de guider le

Pour plus d'information sur Objecteering 6.1, pour télécharger Objecteering **Free Edition** ou Objecteering **Enterprise Edition**, rendez-vous sur : [www.objecteering.com](http://www.objecteering.com)

UML2 BPMN  
C# MDA SysML  
Java SQL  
J2EE SOA  
C++

# Objecteering

Venez découvrir

## Objectteering Scope Manager

à Paris le 18 septembre  
**Séminaire gratuit**

Inscrivez vous sur  
**www.objectteering.com**

\* Vos projets méritent un outil

[www.objectteering.com](http://www.objectteering.com) - Tél. : 01 30 12 16 60 - [sales@objectteering.com](mailto:sales@objectteering.com)

Les écrans tactiles existent depuis plus de 20 ans. Régulièrement, il y a des tentatives pour démocratiser cette interface. Microsoft l'avait tenté avec le projet Origami ou encore le TabletPC. Deux échecs. Pourtant la technologie tactile existe et est mature, il n'y a qu'à voir les bornes SNCF. Cependant, il manquait un déclic.

# La RÉVOLUTION du MULTI-TOUCH





© Photo Microsoft

Et ce fameux déclic est venu il y a un an avec Apple et son iPhone. La Pomme rendait disponible à tout le monde la technologie multi-touch, qui va bien plus loin que le simple tactile. Il s'agit réellement de piloter les applications, les fonctions au doigt, enfin, à plusieurs doigts ! Et depuis, éditeurs et constructeurs se ruent sur le multi-touch : HTC, Nokia, HP, Microsoft. Tout le monde veut sa couche multipoint ou proposer un ordinateur " touch ", mais avec des niveaux fonctionnels parfois décevants ou limités. Et les Français ne sont pas en reste avec des projets chez Intuilab, Numicom ou encore dans le e-lab de Bouygues. Quelques projets open source existent. Mais nous n'en sommes qu'au début de la vague.

Les premiers PC réellement multi-touch vont commencer à être vendus. Apple propose des fonctions multi-touch sur les portables, via le trackpad, mais pas encore au niveau écran, contrairement à HP qui propose le TouchSmart. Bien des questions demeurent sur l'accueil des utilisateurs, les applications adaptées, et bien entendu, le support des développeurs sur ce type de terminaux. Car si pour le iPhone, on dispose d'un SDK et surtout d'un Cocoa Touch, peu d'éditeurs / constructeurs proposent de telles API pour leurs produits ! Et c'est là une faiblesse du marché actuel qui se cherche. Comme vous le verrez, nous ne sommes qu'au début de la révolution multi-touch ! La prochaine conférence PDC (Micro-

soft) en octobre, devrait fournir un début de réponse quant aux ambitions de l'éditeur sur cette technologie dans Windows Seven et CE. Linux ne restera pas en retrait, ni les autres systèmes. Vous comprendrez la différence entre un écran tactile et le pur multi-touch, car il y a parfois un fossé fonctionnel et si l'écran multi-touch est toujours tactile, l'inverse n'est pas vrai. Mais finalement comment cela fonctionne-t-il ? Quelles sont les architectures et les offres du marché ? Quel intérêt pour l'utilisateur ? Et la question que vous attendez tous : *"et pour le développeur, comment cela passe-t-il ?"* Toutes les réponses dans ce dossier !

■ Dossier réalisé par François Tonic



# Comment ça marche ?

Pour bien comprendre comment fonctionne le multi-touch (ou multi-point en français), parlons maintenant d'architecture. Dans cette technologie, il y a deux éléments nécessaires et incontournables : le matériel et le logiciel (middleware, API / SDK, applications).

■ François Tonic

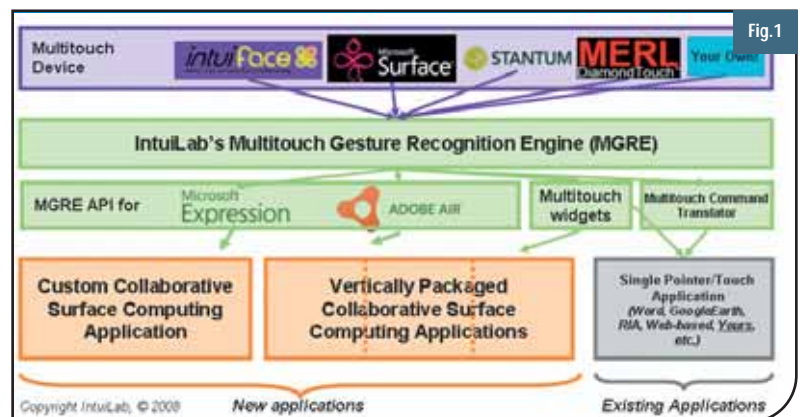
**i**l faut se méfier de l'appellation parfois abusive des constructions autour du multi-touch. Le iPhone, par exemple, est un vrai multi-touch avec un écran capacitif. Souvent, vous trouverez des PC ou des smartphones dits multi-touch qui seront en réalité des terminaux monotouch poussés au multi-touch, mais aux fonctionnalités limitées par rapport à ce que peut offrir un vrai multi-touch. C'est une confusion entretenue volontairement.

## L'architecture du multipoint

### • Le matériel

Pour bénéficier du multi-touch, il faut posséder un matériel adapté. Et selon l'écran, on parlera de monotouch, de dualtouch, ou de multi-touch, et il y a encore des subtilités dans les capacités de l'écran, comme le capacitif. Il existe différents types d'écrans prêts au multipoint.

L'écran de l'iPhone n'est pas un LCD classique. Il possède plusieurs couches ayant un rôle dans le multipoint. Basiquement, il s'agit de pouvoir détecter sur l'écran la présence (la pression) d'un ou plusieurs doigts sur la surface. Surtout, il faut pouvoir très précisément localiser le ou les doigts. Dans le cas de l'iPhone, nous sommes en présence d'un écran capacitif car il détecte les contacts sur l'écran par un courant électrique passant dans une couche de l'écran (couche transparente conductrice). D'autres types d'écrans utilisent la lumière pour détecter un objet. C'est ce signal électrique que le système va utiliser pour exécuter les gestes pris en charge. Il sait détecter et utiliser deux doigts, nous sommes donc dans



du dualtouch. Actuellement, le smartphone ne va pas au-delà car ce n'est pas l'objectif de ces terminaux. Ensuite, vous avez des dalles de grande taille utilisées, par exemple, par Microsoft Surface, Stantum, Intuilab [Fig.1]. Là aussi, plusieurs technologies sont possibles. Dans le jargon actuel, on appelle ce marché le " **Surface Computing** ". Ce marché est à différencier du PC normal et des téléphones. Pour ces surfaces, en général, la dalle possède un système infrarouge et une ou plusieurs caméras (infrarouge) pour détecter les

objets posés sur la surface. Cela permet de définir les X / Y (les coordonnées) pour pouvoir les envoyer au middleware, puis dans les applications. D'autre part, pour garantir une bonne reconnaissance des objets (les placements et déplacements) il faut pouvoir " scanner " la surface constamment. Par exemple, dans la solution du français Intuilab, la caméra balaie la surface 60 fois par seconde. Cependant, certaines tables comme Surface, sont sensibles à la lumière, or cette sensibilité rend instable l'utilisation du multipoint et perturbe les

performances et les reconnaissances des gestes. Dans ce cas, bien souvent, l'infrarouge est utilisé en projection dans la table (sous la dalle) alors que les caméras servant à repérer (et uniquement à détecter) sont moins sensibles à la lumière.

#### • Utiliser des objets sur une surface

On se souvient encore des démos de Surface avec des téléphones et objets divers que l'on pose et que la table reconnaît. Comment est-ce possible ? Surface utilise des tags que la caméra reconnaît. Ces **tags** lui sont spécifiques. L'autre solution, retenue notamment par Intuilab, est de passer par des objets contenant un tag **RFID**. Le e-lab de Bouygues passe par la spécification **DLNA** (voir le reportage dans ce dossier). Bref, il n'y a rien de magique là dedans et la gestion du tag doit être intégrée dans le système multi-touch et dans les applications.

#### • Les gestes [Fig.2]

Contrairement à ce que l'on pourrait penser, le multipoint n'est pas permis. Et cette limitation est simple à comprendre. Pour fonctionner, le système logiciel du multipoint possède un moteur de gestes (**Gesture Engine**) intégrant une série de gestes. Ce sont les actions que l'utilisateur peut faire. Par exemple : agrandir/réduire une fenêtre ou une image, ouvrir une

application, faire défiler le contenu d'un fichier, afficher le clavier virtuel, supprimer un objet, etc. Tous ces mouvements sont enregistrés dans le moteur et si un mouvement n'est pas prévu pour le système ou l'application, le geste n'aura aucun effet. Ainsi Intuilab, Apple, Microsoft, HTC, etc. possèdent une bibliothèque de gestes. La prochaine étape est de pouvoir définir soi-même un geste et de l'inscrire dans le moteur. Cette possibilité permet alors de mettre dans une application des gestes non inclus en standard par le système. Là se situe le travail du développeur et du designer qui doivent repenser le fonctionnement et l'interface.

#### API, SDK, outils

Vous me direz " *bon ok, c'est bien beau, mais comment je code ?* ". Bien entendu, les terminaux et dalles multi-touch se codent très bien, à condition d'accéder aux API, SDK, drivers et autres librairies ! Dans le cas de l'iPhone, Apple propose depuis quelques mois un SDK iPhone 2.0. Le système MacOS X de l'iPhone possède une couche spécifique, le Cocoa Touch. Il se code en Objective-C (langage historique d'OS X). Cette couche permet d'accéder aux fonctions de l'accéléromètre, au multipoint, à la caméra, etc. (reportez-vous à l'article programmation iPhone de ce dossier).



Fig.2

Côté Microsoft, il existe aussi un SDK Surface mais celui-ci n'est pas accessible librement. Cependant, dans quelques semaines, le SDK multi-touch pour Windows Seven sera disponible. On peut penser qu'il sera prochainement intégré dans .Net. Intuilab possède aussi son jeu d'API, non accessibles librement. Celles-ci se codent en .Net (C#, WPF, XAML) ou en AIR avec MXML, ActionScript. Le constructeur français supporte les deux environnements, ce qui est un avantage non négligeable. Et là, le développeur et le designer utiliseront les outils Visual Studio, la gamme Expression, FlexBuilder, etc. Les constructeurs de dalles proposent des API plus ou moins structurées. Cependant, dans le cas d'écran tactile (monotouch), bien souvent, une surcouche au système est rajoutée et dans ce cas, on ne dispose pas d'API, car le système gère par défaut. Ainsi dans le projet SmartCooking (Imagine-Cup 2008), l'équipe française a juste rajouté la couche de gestion tactile de l'écran sur Windows, évitant ainsi tout codage en dur.

Le plus difficile dans le développement multipoint est la bonne gestion des événements. Si on regarde la chaîne de conception Intuilab avec IntuiKit, le Français propose une approche itérative des projets dans laquelle le client est en relation constante. Outre la

## Les marchés visés

Le marché du multi-touch est très segmenté. Le Surface Computing (à large écran) vise pour le moment les professionnels : en showroom par exemple ou dans un cadre purement professionnel (ex. : en CAO). Le Surface Computing vise bien entendu le grand public notamment en usage dans un magasin, une banque, etc. Le Surface Computing est un argument pour faire revenir le client car il n'en possède pas chez lui ! Et pour la maison justement ? Actuellement, les terminaux mobiles démocratisent l'usage du multi-touch. Sur un PC, il faut en avoir l'utilité et surtout une bonne implémentation et non une simple surcouche limitée à quelques

applications, au risque de devenir frustrant. Mais il ne faut pas oublier que le Surface Computing vise tout de même le travail collaboratif : partage de documents, travail sur des documents, etc. Quid des applications actuelles ? Intuilab permet de simuler clavier et souris, ce qui rend possible la gestion des applications comme Word, Google Earth sans modifier l'application ! Le moteur interne transformant les actions des doigts en action clavier – souris ! Pour Intuilab, il s'agit aussi d'informatiser des processeurs humains que l'on automatisait très mal comme une gestion aérienne, la conduite d'un train, etc.

disponibilité du kit en WPF et AIR, il s'agit de rassembler des composants provenant de différentes sources :

- graphe de scène en XML, XAML, MXLML : c'est la modélisation de l'interface
- un système de message (middleware)
- un graphisme SVG, Illustrator, XAML,
- la gestion du comportement (data-flow, règles, gestion des états)
- les interfaces utilisateurs (voix, tactile, vidéo, etc.)

Tout cela aboutit à la génération de l'application finale. Mais un des credo est la stricte séparation de l'interface et du code. Techniquement, coder une application multi-touch n'est pas plus compliqué que de bien coder une application multithreadée.

## Une question d'interface ?

Cependant, attention, un défi est à relever : celui de l'interface utilisateur. Pour Vincent Encontre (directeur général d'Intuila), il s'agit de sortir du concept **WIMP** (Window, icon, menu, pointing device), bref de l'interface telle qu'on la connaît aujourd'hui. Il s'agit de sortir aussi du concept de **GUI** (interface graphique utilisateur). Le rôle du designer, de l'ergonome se révèle primordial. Et la nouvelle génération d'outils de développement va dans cette direction. Cela prend tout son sens dans le Surface Computing mais quand le PC aura des capacités étendues de multi-touch, l'interface sera remise en cause tout en mêlant l'approche WIMP. Et on aboutira à la

## Java et le multi-touch

Pour le développeur Java, que peut-on faire pour le multi-touch ? Sun, durant la dernière JavaOne, a montré les capacités de JavaFX à faire du multi-touch, mais reste bien discret sur les capacités réelles de la plate-forme. La session " Building own multi-touch interface with Java and JavaFX " montrait comment assembler sa propre table, avec une web cam pour ensuite passer par une capture d'images, via Java Media Framework, et analyser les images pour déterminer la

position des doigts pour ensuite permettre à JavaFX sur l'application d'agir en conséquence... Pour le moment, il n'y a rien de réellement concret sur le multi-touch dans JavaFX. L'autre solution est de passer par le framework OpenInterface. Il s'agit d'un framework conçu en C++ pour pouvoir créer et gérer des applications supportant les interfaces multimodales (doigt, voix, image, etc.). Le noyau reste en C++ et la communication avec un lecteur mp3,

une Wiimote, etc. via des données XML. Il s'agit d'un projet particulièrement intéressant et open source. Sur le multi-touch pur, l'open source reste actuellement à la traîne. Il existe le projet MPX (Multi-pointer X) tournant sous Linux mais avec le départ du principal développeur, le projet semble aujourd'hui ralenti, voire mort alors que les premières versions étaient plutôt prometteuses. Il utilise un X Windows Server et un simple écran capacitif.

conception d'applications allant indifféremment pour une utilisation multi-point ou non. Pour l'utilisateur, tout se fera en transparence et le développeur codera et compilera une seule application ! Ainsi **FlexBook** (pour tourner les pages " naturellement ") est disponible à la fois en technologie clavier - souris et en multi-touch... Et son utilisation est bluffante, je vous le garantis !

Pour compléter ce dossier, le site de Programmez ! vous propose une collection complète de vidéos sur le multi-touch : [www.programmez.com](http://www.programmez.com)

## Stantum : l'autre français du multi-touch

Notre pays possède une solide expérience dans le multi-touch. Notre autre grand acteur est la société Stantum basée à Bordeaux. Elle propose ainsi un kit de développement comprenant une dalle LCD, les pilotes multi-plates-formes, les connecteurs USB, les interfaces adéquates pour coder, utiliser et récupérer les données de la dalle dans ses applications.

## Projet Majook (Numicom)

Le français Numicom et le Game Developer Laboratory de l'Epitech travaillent ensemble sur le projet Majook. Il s'agit d'une plate-forme interactive capable de reconnaître des objets communicants. Cela ressemble à du Surface Computing tel qu'on l'entend aujourd'hui. Basiquement, le plateau détecte la nature et la position d'un objet posé sur lui et interagit ensuite avec lui. Si aujourd'hui il se limite à une relation avec un PC en sans fil, demain, ce

sera une communication avec une télévision, un téléphone, etc. Il ne s'agit pas de multi-touch mais les applications possibles sont là aussi immenses, notamment pour le jeu. Actuellement, le kit de développement et les pilotes sont en pleine phase de développement. Numicom définit le projet comme une expression de l'interface tangible. Il s'agit aussi de réduire les coûts de fabrication et donc de le proposer à un prix acceptable. Actuellement le



SDK est en C++ mais d'autres langages sont à l'étude. L'IDE utilisé est Eclipse.



# ihm

EN TOUTE SIMPLICITÉ !

Votre application de gestion multilingue  
avec plus de 100 vues de 20 types  
différents, en DHTML/Ajax,  
en Swing ou en plugin Eclipse,  
connectée à un SGBD et un bus JMS...

“ Il vous faut  
combien de temps  
pour la réaliser ? ”

**NOUVELLE  
VERSION!**  
**LEONARDI  
V4.0  
OPEN SOURCE**

Si votre réponse est moins d'une  
semaine, inutile de vous rendre sur  
notre site, ni de télécharger la  
version gratuite de LEONARDI,  
sinon il est temps de passer  
à la vitesse "Model-Driven"...



# Développez pour l'iPhone !

Concevoir une interface tactile représente une évolution beaucoup plus profonde que de simplement remplacer la souris ou le stylet par le doigt. L'ensemble du système doit être repensé, et c'est ce qu'Apple a réalisé avec l'iPhone.

**C**et article vous propose de parcourir les différences essentielles que suppose ce changement dans l'interface utilisateur, puis de décrire les objets utilisés par iPhone-OS pour mettre en œuvre la gestion du Multi-Touch. Ces concepts sont illustrés par un exemple de code permettant de visualiser l'évolution des points de contact des doigts à la surface de l'écran de l'iPhone. Un ensemble mature et cohérent

## Souche commune de Mac OS X

L'interface Multi-Touch de l'iPhone est un développement totalement nouveau pour la Pomme. Cependant, il s'appuie sur des briques logicielles éprouvées ayant déjà démontré leur efficacité. Il faut en effet se souvenir que les premières stations NeXT qui faisaient tourner Objective-C et l'ancêtre de Cocoa il y a presque vingt ans, disposaient d'une puissance de calcul inférieure à celle d'un iPhone !

L'interface Multi-Touch de l'iPhone reprend donc les éléments fondamentaux suivants :

- Le pattern MVC qui permet de bien distinguer le modèle de données, la présentation et les actions modifiant les données.
- Le langage Objective-C dans sa version 2.0 avec, en particulier, la notion de propriétés qui simplifie la réalisation des accesseurs sur les variables d'instance. Les propriétés sont largement employées dans les API d'iPhone-OS.
- Le framework Cocoa avec ses classes fondamentales permettant de manipuler les structures de base (telles que chaîne de caractères, listes, etc.), les objets de tracé graphique, la hiérarchie de vues pour composer le contenu des fenêtres et les contrôles essentiels pour l'interface utilisateur : champs de saisie de texte, barres d'avancement, etc.

Il reprend également Core Animation et ses fonctions de transition ou de transformation animée des propriétés des éléments de l'interface utilisateur. Cette technologie a été initialement développée pour l'iPhone parce qu'elle donne tout son sens aux actions de l'utilisateur. Il a parallèlement été intégré à Mac OS X dans sa version 10.5.



## Particularités

L'iPhone est un appareil ultra-mobile pour lequel le système entier a été optimisé en prenant en compte à tout instant :

- la taille de l'écran
- la quantité de mémoire disponible
- l'économie de la batterie

A ces contraintes s'ajoutent des possibilités nouvelles liées à la mesure des accélérations selon 3 axes, la géolocalisation, la téléphonie et la prise de vue. Ces deux dernières fonctions sont absentes de l'iPod Touch. Les frameworks de développement d'iPhone-OS s'articulent pour proposer toutes ces fonctionnalités au travers d'une intégration homogène et cohérente. C'est cette homogénéité et cette cohérence que l'utilisateur perçoit sous la forme d'une grande simplicité d'utilisation. L'interface tactile constitue la partie la plus évidente de cet ensemble.

## Ce qui change avec une interface Multi-Touch

### La précision

Une première évidence : l'extrémité d'un doigt n'a pas la même finesse qu'un pointeur. Cette taille plus imposante empêche de désigner un point précis : toute la surface du doigt est susceptible d'être prise en compte par les capteurs. En s'approchant de la surface sensible, le doigt masque également au regard une zone plus large que le point d'impact réel. Ces deux facteurs d'imprécision impliquent un dessin spécifique pour l'interface utilisateur. Nous aurons l'occasion d'y revenir.

### La multiplicité

D'autre part, l'interface de l'iPhone n'est pas seulement tactile : elle est " Multi-Touch ". Cela signifie que les capteurs sont sensibles à plusieurs points d'impact simultanés. Nous qui sommes habitués à la pointe unique de la mine d'un crayon, d'un stylet ou de la souris, il s'agit d'une petite révolution dans la manière de désigner quelque chose. Nous touchons là, dans tous les sens du terme, la

plus grande originalité de ce mode de communication qui peut impliquer la main entière et non pas seulement l'index. Comme la peau d'un être vivant, la surface sensible de l'iPhone sait distinguer la signification différente entre la main qui se pose et le doigt qui appuie.

### Le geste a un sens

Enfin, cette même main qui se pose peut transmettre un message totalement différent à son récepteur selon que le premier contact se poursuit par l'effleurement de la zone d'impact ou bien par une simple prolongation d'un contact très localisé. C'est ainsi l'évolution dans le temps des points de contact qui va définir l'opération réalisée sur une même zone de toucher.

### Expérience utilisateur

Dans un même geste, on désigne donc, à la fois, la cible et une action. Le principe de communication avec un pointeur utilise la règle suivante : je sélectionne (habituellement avec un clic) puis j'applique une action (généralement avec un menu déroulant). Avec l'iPhone il n'y a plus cette distinction entre sélection puis action, donc pas d'éléments " sélectionnés " à l'écran mais la notion de UIResponder que nous détaillerons plus loin. Cette caractéristique essentielle a conduit Apple à développer tout un ensemble de contrôles spécifiquement adaptés et à proposer des modes de navigation intuitifs.

## L'implémentation de l'interface Multi-Touch

### Gestion des événements

Les principales actions de l'utilisateur sont modélisées, de manière assez classique, avec la notion d'événement. Un événement encapsule les contacts des doigts sur la surface de l'écran à un instant donné. Les événements sont créés par le système puis placés dans une file d'attente que l'application vient régulièrement consulter. Lorsque l'application récupère un événement, elle parcourt la hiérarchie d'inclusion des vues pour déterminer celle qui contient le point de contact. Cette vue devient alors le premier élément dans la chaîne des répondeurs. Cette chaîne est constituée d'objets descendant de la classe UIResponder. Chaque élément de cette chaîne peut :

- gérer lui-même l'événement et terminer son traitement (c'est le cas de la plupart des contrôles qui transforment l'événement en une action).
  - décider que l'événement ne le concerne pas et le transmettre à l'élément suivant dans la chaîne des répondeurs.
- Le répondeur suivant une vue dans la chaîne est son contrôleur. Le répondeur suivant le contrôleur d'une vue est la vue contenant cette vue, ou la fenêtre, en cas de vue racine. Le répondeur suivant la fenêtre est l'application. Pour gérer un événement, un répondeur implémente simplement au moins l'une des méthodes suivantes :

- touchesBegan:withEvent:
- touchesEnded:withEvent:
- touchesMoved:withEvent:
- touchesCancelled:withEvent:

### L'objet de type UITouch

Un événement contient la description des contacts des doigts sur la surface de l'écran à un instant donné. Cette description est réalisée par la classe UITouch. Un événement est donc constitué d'un ensemble d'objets de type UITouch. Chaque objet UITouch correspond ainsi au contact d'un doigt sur l'écran. L'évolution du contact d'un doigt sur l'écran se traduit par une succession de phases : début, fin, mouvement, stationnaire et annulé. Chaque changement de phase pour un contact de doigt donne lieu à la création d'un événement. En plus de la phase, un objet UITouch contient les coordonnées du point de contact et la référence sur la vue et sur la fenêtre contenant ce point.

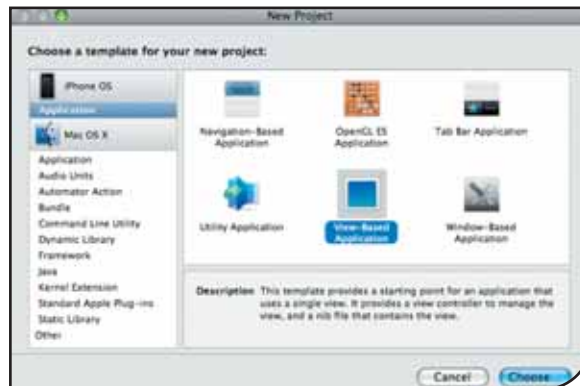
### Interpréter un geste

Interpréter un geste consiste donc à suivre l'évolution des phases et des coordonnées du point de contact. En particulier, un mouvement de balayage horizontal peut être repéré par une évolution relativement limitée de la coordonnée verticale en comparaison de l'évolution de la coordonnée horizontale.

## Visualisation des objets UITouch

### Les objets créés par défaut par Xcode

Nous allons réaliser un exemple très simple permettant de visualiser les objets UITouch transmis à une application. Pour cela nous utilisons le template " View-Based Application " proposé par Xcode pour iPhone-OS. En indiquant le nom " Programmez-Touch ", ce template crée pour nous l'ensemble des objets nécessaires :



- La classe Programmez\_TouchViewController qui gère les interactions de la vue principale avec le modèle de données qu'elle représente.
  - La classe Programmez\_TouchAppDelegate qui contrôle l'application elle-même avec des fonctions telles que la création de la fenêtre principale et du Programmez\_TouchViewController.
  - Des documents de type .xib permettant de décrire graphiquement les propriétés des objets de l'application.
- Nous allons personnaliser quelques-uns de ces éléments pour créer notre application en suivant simplement le pattern MVC.

### Modèle touchViews

Nous commençons par définir notre modèle de données. Dans notre exemple, il s'agit de référencer les instances



UITouch que reçoit la vue. Pour cela nous utilisons un dictionnaire de type `NSMutableDictionary`, nommé `touchViews`, permettant d'associer une valeur à chaque instance `UITouch` reçue par l'application. La valeur que nous associons est l'image d'une empreinte digitale dont les coordonnées dans la vue reflètent celles de l'objet `UITouch`. Chaque image d'empreinte digitale est réalisée par une instance de type `UIImageView` placée comme sous-vue de la vue principale.

## Vue UIView

Xcode a déjà créé la vue principale pour nous. Elle se trouve dans le document `Programmez_TouchViewController.xib` que nous ouvrons avec Interface Builder. Tous les paramètres par défaut sont correctement positionnés à l'exception de l'attribut `multipleTouchEnabled` qui ne lui permet pas de recevoir plusieurs contacts simultanés s'il n'est pas coché.

## Contrôleur

### Programmez\_TouchViewController

Le contrôleur concentre l'essentiel du comportement de notre application. On peut distinguer quatre grandes fonctionnalités :

- La création et la libération du modèle `touchViews`
- L'ajout d'une nouvelle entrée dans le dictionnaire lors d'un événement `touchesBegan:withEvent:`
- La suppression de l'entrée correspondant à un événement `touchesEnded:withEvent:`
- Le déplacement de l'image de l'empreinte digitale pour un événement `touchesMoved:withEvent:`

Nous allons maintenant détailler le code réalisant ces fonctionnalités en Objective-C.

## Le code de la classe

### Programmez\_TouchViewController

#### Gestion de touchViews

`touchViews` est défini comme une variable d'instance du contrôleur :

```
@interface Programmez_TouchViewController : UIViewController {
    NSMutableDictionary * touchViews;
}

@end
```

Le cycle de vie de `touchViews` est extrêmement simple : `touchViews` est créé en même temps que la vue à laquelle il est associé et il est détruit avec le contrôleur. La capacité initiale du dictionnaire est fixée à 5 ce qui correspond aux contacts des 5 doigts d'une main !

```
- (void)viewDidLoad {
    [super viewDidLoad];
    touchViews = [[NSMutableDictionary alloc] initWithCapacity:5];
}

- (void)dealloc {
    [touchViews release];
}
```

## Et Windows 7 ?

C'est en octobre prochain que la première version du SDK multi-touch pour Windows 7 (le prochain Windows) doit sortir des bureaux de Microsoft. Le système incorporera une couche complète de multi-touch reprenant un certain nombre de fonctions de Microsoft Surface.

On bénéficie ainsi d'une manipulation à deux doigts sur les images, d'un outil graphique adapté (Paint) avec possibilité d'utiliser tous ses doigts ! Les fonctions de cartographie seront aussi présentes, la navigation système, etc. Le kit de développement sera sans doute disponible en langage .Net et en C++ et intégré (plus ou moins immédiatement avec .Net et Visual Studio). La partie mobile intégrera aussi la même technologie. On devrait même pouvoir disposer d'un moteur de gestes (avec possibilité de définir ses propres gestes ?). Reste à voir comment va se présenter le modèle de développement et sa cohabitation avec les applications actuelles. Il ne fait guère de doute que WPF sera largement utilisé par cette nouvelle interface.

D'autre part, un des défis est d'adopter des gestes standard car s'il faut apprendre des gestes spécifiques à chaque plate-forme ou système, l'utilisateur sera perdu. Mais les éditeurs auront-ils la volonté de s'entendre sur un marché aussi prometteur ?

Cette disponibilité en standard boostera sans nul doute le marché des écrans multi-touch PC. Dell a d'ailleurs récemment annoncé de nouveaux modèles.

■ F.T.

```
[super dealloc];
}
```

Remarque : Si le cycle de vie avait nécessité un accesseur ou un modificateur pour `touchViews`, on aurait pu définir la propriété correspondante pour générer automatiquement les méthodes supplémentaires `touchViews` et `setTouchViews`.

## Gestion des événements

Le contrôleur étant le successeur immédiat de la vue principale dans la chaîne des répondeurs, il reçoit les événements que la vue n'a pas traités. C'est donc lui qui implémente `touchesBegan:withEvent:`, `touchesEnded:withEvent:` et `touchesMoved:withEvent:`. Pour chacune de ces méthodes, l'ensemble des objets `UITouch` de l'événement est parcouru par une boucle `for`. La fonction réalisée avec l'objet `UITouch` dépend du type d'événement :

- Un événement `touchesBegan` correspond à l'apparition d'un nouveau contact sur l'écran et donc à l'ajout d'une empreinte.
- Un événement `touchesEnded` correspond à la fin d'un contact sur l'écran et donc à la suppression d'une empreinte.



Le Meilleur du Test de Performance Web en toute simplicité et au meilleur rapport qualité-prix



- Développement automatisé de Cas-Tests et Test de Charge
- Rapports d'analyse des Tests de performance et de charge
- Analyse des Serveurs et de l'impact sur les performances
- **nouveau** Interface utilisateur et Rapports en Français

Découvrez comment Web Performance Suite peut faire gagner vos applications Web en Qualité, Fiabilité et Performance

Evaluation gratuite de Web Performance Suite sur <http://www.kapitec.com/Pub/WP?id=111>

Bénéficiez de notre assistance technique pendant les 15 jours d'évaluation.



Power Your Web Projects

Tél.: 05 34 27 90 03  
[sales@kapitec.com](mailto:sales@kapitec.com)

## Comment devenir développeur iPhone ?

L'inscription se fait en ligne sur le site <http://developer.apple.com/iphone/program/>. Le premier niveau d'inscription (gratuit) permet d'accéder à l'ensemble de la documentation et de télécharger les outils rassemblés dans le SDK. Le niveau payant individuel (79€) donne accès à un portail spécifique sur lequel il est possible de créer les certificats nécessaires pour constituer les profils de provisionnement. Un premier profil de provisionnement, le profil développeur, peut être installé sur un iPhone ou un iPod Touch et autorise les applications signées par le certificat correspondant à fonctionner sur ces appareils. Un second profil, le profil de distribution, permet à une application signée avec le certificat correspondant d'être distribuée sur l'App Store. Le niveau payant "entreprise" permet de gérer des équipes de développeurs ayant chacun leur certificat individuel.

- Un événement `touchesMoved` correspond au déplacement d'un contact sur l'écran et donc au déplacement d'une empreinte.

Dans tous les cas, la propriété `view` du contrôleur donne accès à la vue principale qui définit le système de coordonnées.



### Ajout d'une empreinte

L'image de l'empreinte digitale est un document PNG qui est le format graphique que l'iPhone gère de manière optimisée. Le `retainCount` de `fingerPrintView`, implicitement incrémenté par la séquence `alloc/init`, est décrémenté par un appel à `release` lorsque la vue a été insérée dans le dictionnaire et dans la hiérarchie des vues. De cette manière, le retrait de cette même vue de la hiérarchie, puis du dictionnaire, provoquera automatiquement sa libération. Notez que l'ajout dans le dictionnaire se fait avec un appel au niveau Core Foundation. En effet, la méthode `setObject:forKey:` de `NSMutableDictionary` réaliserait une copie de l'objet `UITouch` avant de l'insérer, or c'est bien la référence sur l'objet inclus dans l'événement que l'on souhaite utiliser comme clef pour le dictionnaire.

La valeur de la propriété `center` de la vue `fingerPrintView` est positionnée aux coordonnées du point d'impact.

```
(void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event {
    for (UITouch *touch in touches) {
        UIImage *fingerPrintImage = [UIImage imageNamed:@"finger-print.png"];
        UIImageView *fingerPrintView = [[UIImageView alloc] initWithImage:fingerPrintImage];

        CFDictionaryAddValue((CFMutableDictionaryRef)touchViews, touch, fingerPrintView);
```

```
fingerPrintView.center = [touch locationInView:self.view];
[self.view addSubview:fingerPrintView];

[fingerPrintView release];
}
```

### Suppression d'une empreinte

Pour supprimer l'empreinte associée à un contact qui disparaît, on commence par récupérer dans le dictionnaire la vue de l'empreinte digitale correspondante, puis on la retire de la hiérarchie des vues et enfin du dictionnaire.

```
(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event {
    NSLog(@"touchesEnded, touches: %@", touches);
    for (UITouch *touch in touches) {
        UIImageView *fingerPrintView = [touchViews objectForKey:touch];
        [fingerPrintView removeFromSuperview];
        [touchViews removeObjectForKey:touch];
    }
}
```

### Déplacement d'une empreinte

Pour refléter le déplacement d'un point de contact sur l'écran, on récupère dans le dictionnaire la vue de l'empreinte digitale correspondante, puis on modifie sa propriété `center`.

```
(void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event {
    for (UITouch *touch in touches) {
        UIImageView *fingerPrintView = [touchViews objectForKey:touch];
        fingerPrintView.center = [touch locationInView:self.view];
    }
}
```

### Test de l'application

Le projet complet peut-être téléchargé sur le site <http://osx-prog.com//programmez>. Le simulateur permet alors de tester l'application en créant un point de contact avec le pointeur du Mac. Un second point de contact symétrique peut être obtenu avec la touche "alt". Cependant, pour obtenir plusieurs points de contact indépendants, il faut impérativement utiliser un appareil spécialement préparé avec Xcode et un profil de développeur (voir encadré).

#### ■ Etienne Vautherin

ancien Technology Evangelist chez Apple, il est chef de projet chez Orange Labs et auteur de "Mac OS X Programming" chez Dunod.



# Les nouvelles IHM à base d'interface tactile

Lors des derniers TechDays 2008 de Microsoft, le e-lab de Bouygues a présenté une table tactile équipée d'une technologie multi-touch. Pour en savoir plus, nous avons rencontré l'équipe Nouvelles Technologies du e-lab qui a réalisé ce prototype.

■ François Tonic



Le e-lab est un petit centre de Recherche et Développement, rattaché à la maison mère du groupe Bouygues. Ses missions consistent à détecter les entreprises innovantes, les start-up, et à les mettre en relation avec les responsables métiers des différentes filiales. Parfois le produit que propose la start-up n'est pas assez mature. Dans ce cas, le e-lab accompagne et développe avec la start-up un produit qui pourra ensuite correspondre aux besoins des métiers du groupe. Le e-lab a aussi la charge de l'animation de l'Innovation via la Filière Innovation Bouygues. Cette filière composée de 200 managers de l'innovation et des nouvelles technologies, se réunit 3 fois par an, afin d'échanger sur les process d'innovation et de conforter ses idées avec d'autres industriels, invités par le e-lab.

Le e-lab est composé de deux équipes :

- Une équipe optimisation qui possède des compétences en mathématiques en plus de son expertise en Systèmes d'Informations,

- Une équipe Nouvelles Technologies qui possède des compétences en développement, en électronique et en mécanique, ainsi qu'en assemblage de technologies diverses,

Une des principales missions de l'équipe Nouvelles technologies qui a réalisée la table tactile, est de réali-

ser des assemblages de technologies, logiciels ou matériels, permettant de résoudre des problématiques métiers.

## Observer, considérer

Une autre de ses missions est d'être en contact avec le monde de la recherche et donc avec l'extérieur du groupe Bouygues. C'est dans le cadre de cette mission que le e-lab a décidé de présenter en 2008 sa table tactile aux TechDays, mais aussi au Remix 2008 de Microsoft. Cette visibilité permet au e-lab de rencontrer des développeurs, d'échanger avec des experts en ergonomie ou des designers sur des salons, lors de séminaires, au sujet de ses différentes missions. Chaque année, le e-lab rencontre une cinquantaine de jeunes pousses ou petites PME, qu'il accompagne dans le groupe, ou bien met en contact avec les bonnes personnes dans les différentes filiales du groupe : " Nous avons des relations directes avec ces start-up. Elles ont du potentiel, même si ce n'est pas toujours mature. On les conseille, on les suit et on les aide. " explique Guillaume Rochart (ingénieur de recherche). Pour finir, le e-lab est aussi un centre de Recherche. Grâce à ses liens avec le CEA, l'INRIA, des universités ou des écoles d'ingénieurs, le e-lab travaille sur des projets de Recherche, en relation

avec les métiers du groupe. Il est parfois amené à travailler sur des sujets qui n'ont pas encore de réalité industrielle, mais cela fait partie de sa mission de défrichage des technologies.

## Nouvelles interfaces

Une des certitudes du e-lab est que la maison de demain sera complètement connectée et cela passe par plusieurs éléments :

- le serveur média tel que l'on peut en avoir actuellement chez Microsoft avec le Home Server,
- le player : écran ou tout autre " display ", chaîne Hifi IP (cf. Sonos),
- la commande média : télécommande universelle, table tactile, téléphone mobile.

## Les Interfaces tactiles multipoint : une solution d'avenir accessible

La table tactile du e-lab ressemble un peu au concept de Microsoft Surface ou au produit Intuilab, même si dans

## DLNA

Digital Living Network Alliance. L'objectif de cette alliance est de définir un standard pour tous les appareils électroniques, informatiques, domestiques pour communiquer, échanger, se reconnaître. Par exemple, la PlayStation l'est déjà. La table du e-lab supportant cette spécification peut facilement reconnaître et communiquer avec ces appareils compatibles.

la pratique il diffère dans les fonctionnalités. La genèse de la table est venue avec la réflexion plus globale sur la maison multimédia, le développement du multimédia à la maison, en gros : la maison numérique ; 6 mois avant la présentation aux Tech-Days, le projet était simplement une évocation. " *Tout a réellement commencé par un séminaire interne ayant eu lieu fin 2007. L'intérêt du projet de la table a été immédiat. L'accueil fut bon* ", indique G. Rochart.

Les aspects tactiles présentent l'intérêt d'avoir une interface plus accessible avec les utilisateurs. Mais dans les faits, la réalisation du projet doit beaucoup au hasard. " *Philippe Caillol était en Asie et est tombé sur une dalle tactile multi-touch. Comme nous avions prévu de travailler sur une table de commande de la télévision, une dalle qui coûtait environ 300 dollars, un prix raisonnable, semblait être un bon point de départ. Aujourd'hui, nous en avons deux exemplaires*", poursuit G. Rochart. Et

l'opportunisme de la situation a continué quelques semaines plus tard aux TechEd de Barcelone, en Novembre 2007, notamment sur la partie WPF qui intéressait l'équipe.

Le plus délicat, comme nous ont confié les membres de l'équipe, est de pouvoir prendre du recul sur les nouvelles technologies, et pour la table, il fallait trouver un intérêt possible : bref quelle utilité cela pouvait avoir. C'est dans ce sens que le projet diffère de l'approche de Microsoft.

" *Nous sommes partis de rien, juste de quelques idées. Il fallait trouver un rapport qualité - prix acceptable pour pouvoir monter une démonstration d'usage à un prix raisonnable. En effet, il ne s'agissait pas de faire un produit mais de monter un projet de recherche pour pousser de nouveaux usages autour de l'accès aux données et aux équipements. Nous avons aussi mené une réflexion en interne sur l'interface utilisateur, notamment auprès des showrooms, pour y présenter des produits d'une autre*

*manière ou encore dans les banques*", indique G. Rochart.

L'architecture est assez simple :

- une dalle tactile infrarouge (constructeur iNexio),
- deux machines (Appel MacMini) pour le système (Windows XP) et les données média,
- un environnement applicatif pour la gestion et les applications.

## Le logiciel en question

Si le choix de Windows et de .Net s'est imposé rapidement, encore fallait-il savoir comment coder l'application. " *Il fallait un framework maîtrisable très rapidement. Je ne pouvais pas du tout modifier .Net pour l'adapter aux besoins. WPF est apparu comme la solution idéale, surtout dans la contrainte de temps, moins de 4 mois !* ", analyse après coup Guillaume Rochart. " *Le fabricant de la dalle ne fournissait qu'un semblant d'API. En fait, seulement une zone mémoire pour récupérer les points détectés sur la dalle. Nous avons reçu un morceau de code pour le faire et nous l'avons adapté à nos besoins. Comme on travaille en 2D, il faut pouvoir récupérer les points, les coordonnées. Pour notre besoin c'est suffisant. Nous faisons du mono utilisateur en multi-touch* " poursuit G. Rochart. " *Bien que nous disposons de deux caméras infrarouge, la dalle ne voit pas si un doigt est caché par un autre (selon l'angle, ndlr). Nous ne faisons pas du capacitif (comme l'i-phone, ndlr)* ", nuance G. Rochart. Pour la partie développement, l'équipe a utilisé Visual Studio 2008, .Net 3.5 (le tout en bêta, car le projet a été codé avant les versions finales). " *Nous sommes partis de zéro !* " s'enthousiasme Guillaume. La table intègre aussi un système de téléconférence qu'il a fallu intégrer (notamment pour la partie protocole SIP). " *Le tout a été compilé sous Vista même si l'application finale fonctionne sous XP* ". Bien entendu, la table sait communiquer avec d'autres objets comme un téléphone ou tout objet ayant le support du DLNA. Les possibilités sont immenses, même si le e-lab avance un peu à l'aveugle car le marché est naissant et en pleine ébullition.

## Questions – réponses avec Guillaume Rochart

**Programmez : Avec le multi-touch, le tactile, qu'est-ce qui change, ou ne change pas, pour l'utilisateur ? Comment apprendre un nouveau paradigme ?**

Guillaume : Je pense que le plus intuitif, c'est l'aspect tactile. On reste très proche des interfaces actuelles mais en facilitant le pointage avec ce qu'on a l'habitude d'utiliser, à savoir nos mains. Pour le multi-touch, par contre, je pense qu'il faudra rééduquer les utilisateurs : il s'agit de nouvelles fonctionnalités. Le but est d'imiter les interfaces réelles (tourner une feuille avec deux doigts, etc.) mais je ne sais pas quelle en est la limite.

**P ! : Et pour les développeurs, comment appréhender une telle technologie ? Faut-il de nouveaux réflexes ? Faut-il repenser l'interface, la manière dont fonctionne l'application ? Ou s'arrête le multi-touch dans une application et est-il adapté à toutes les situations ?**

G : Pour la partie tactile, cela ne change rien ou presque. C'est juste plus simple. Par exemple sur la table, 90% de l'interface se

gère aussi bien à la souris qu'à la dalle tactile. Par contre pour le multi-touch, puisqu'il n'y a pas d'événements gérés nativement dans .Net, il faut tout coder soi-même ; dans mon cas j'ai fait les événements suivants :

- déplacement (la même chose qu'avec une souris mais cela permet d'avoir plusieurs points en références),

- l'ajout/retrait de points (ce qui revient au clic/déclat par exemple, sauf que là ce n'est pas du 0/1 mais du 0 point à n points activés). Je pense que le multi-touch doit être réservé pour des usages où naturellement nous utilisons plusieurs doigts (pour tourner une photo, il est plus simple de le faire avec une main ou au moins deux doigts), ou bien sûr, des actions sur différents objets (puzzle). Pour les styles d'applications, je pense que cela peut s'appliquer à de nombreuses applications : graphismes, grand public, bureautique, industrielle (pilotage d'outils) etc. Un point fort d'une technologie aussi simple qu'une dalle infrarouge est son temps de réaction très réduit.

# L'INFO permanente

- **L'actu** : le fil d'info quotidien des journalistes de Programmez
- **La newsletter hebdo** : abonnez-vous, comme 30 000 professionnels déjà. C'est **gratuit** !

C'est  
**PRATIQUE !**



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

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

## GAGNEZ 300 licences complètes

### Microsoft SQL Server 2008 Edition Standard \*

(valeur équivalente : 1200 euros environ)



Rendez-vous sur [programmez.com](http://programmez.com) ! Inscrivez-vous, et gagnez une des 300 licences de cette version de production.

Réservé aux 300 premiers lecteurs de Programmez et visiteurs de [programmez.com](http://programmez.com)

\* Edition interdite à la revente, version de production avec 1 accès client.

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



## ImagineCup 2008

1ère partie

## Les jeux olympiques du développ

Du 3 au 8 juillet, les développeurs en herbe étaient à l'honneur dans notre pays. Pour la première fois, la France recevait la grande finale d'ImagineCup. Cette année une centaine d'équipes se sont mesurées, affrontées à coup de code, de .Net, de Visual Studio. Et la lutte a été rude pour tenir les délais, affronter le jury et présenter un projet abouti et si possible sans plantage live !



Cette année, la France a dû se contenter d'une médaille d'or avec Challenge IT, et d'une médaille de bronze avec Interface Design. A cela se rajoutent deux prix (awards) pour Well K'home (programme BT Accelerator) et pour JivAd (catégorie Interface Design). En tout, nous avons 6 équipes et 17 étudiantes et étudiants... Nous avons pu constater la présence de nombreuses équipes asiatiques, en particulier de Chine et Singapour, totalisant 6 médailles. L'Europe de l'Est fut fortement représentée par la Russie et l'Ukraine mais aussi la Pologne, totalisant pas moins de 8 médailles dont 2 pour les hongrois et les

ukrainiens. La tension était palpable les premiers jours jusqu'à l'annonce des finalistes. La tête dans le code, les tests, d'autres passaient sous le couperet du jury. Avec plus ou moins de bonheur. Et les heureux finalistes repartaient pour de longues heures de codes, d'ajustement pour les dernières auditions avec les résultats. Les projets ont été d'une grande diversité même si plusieurs avaient une approche commune avec des robots automatiques pour surveiller l'air d'une zone, d'une usine ou encore des automatismes pour mieux gérer sa consommation d'essence ou d'électricité à la maison. Avec un peu d'imagination, Visual Studio,

Expression, .Net et les API Live, on peut faire bien des choses. Beaucoup d'équipes nous ont confié que le plus dur n'était pas la programmation pure mais l'intégration technique entre le logiciel et le matériel et savoir comment utiliser au mieux chaque brique technique. Comme le temps est compté, il faut agir vite, trouver des solutions, parfois juste un bricolage pour éviter un bug, un plantage, ou éviter de montrer au jury la fonction instable. C'est l'occasion aussi pour les équipes de discuter entre elles, de se rencontrer, d'échanger des avis de développeurs sur tel outil, telle API. Ou de trouver des contacts pour un futur job... Tout cela



# ement, à Paris



reste bon enfant, même si l'envie de gagner est présente. L'annonce des finalistes est toujours l'occasion de voir quelques larmes ici ou là. En tout cas, tout le monde repartira avec une expérience renforcée, des souvenirs plein la tête.

Nous avons été particulièrement impressionnés par l'équipe Trail Blazers (Singapour) et son projet EMS permettant, grâce à un robot autonome de s'informer sur la qualité de l'atmosphère, mêlant la communication sans fil, la programmation embarquée et la partie analyse, génération de rapport. Ou encore le projet Vigil (équipe Ignition, Russie), un système pour aider les pompiers à travailler plus

efficacement avec les notions de stratégie anti incendie, une gestion du GPS, la gestion des risques, etc. Il y avait aussi le projet ParkIT (Irlande) pour mieux trouver une place de stationnement sans perdre du temps et donc économiser du carburant. Autre coup de coeur, l'équipe polonaise Aero@PUT avec Ecopterion. Un mini hélicoptère "personnel" pour observer, contrôler l'environnement : sources d'énergie, eau, paysages, etc. Cela est possible grâce à un caméra embarquée et des traitements d'images.

Un prix spécial pour le plus jeune participant : Anthony Platanios du projet grec Project Protasis. Grâce à son énergie, il a conçu et codé

une application PC doté d'un réseau de capteurs, de terminaux mobiles, d'un portail web pour connaître précisément sa consommation énergétique (en kWh) et comment mieux la maîtriser. Et surtout grâce aux capteurs installés et au réseau, on peut via son PC, le portail web, le terminal mobile, contrôler les appareils de la maison. Une belle réalisation ! Comme quoi bien que le thème de l'écologie soit aussi vaste que vague, chaque équipe a pu aller jusqu'au bout de son projet même si certaines applications sont plus convaincantes que d'autres. Rendez-vous en 2009 au Caire pour la prochaine finale mondiale.

■ François Tonic

# Les Français : au sommet

Depuis plusieurs années, les équipes françaises réussissent bien dans ces jeux olympiques de la programmation. Même si elles ont obtenu moins de médailles qu'en 2007, elles ont été récompensées et la qualité de la formation de nos développeurs n'est plus à prouver. Revenons sur plusieurs équipes qui ont su concevoir et coder des projets originaux et d'avenir !

## ECOThink mise sur SimCity

Le jeu est offert sur le CD du numéro !



“ Ecological Tycoon “ est un jeu de stratégie, combinant les aspects “ jeu vidéo “ que l’on trouve dans Sim City ou Civilization, et le développement durable. L’objectif est de rendre ce jeu accessible à tous les joueurs, quel que soit leur niveau, des débutants aux plus aguerris. Avec un double aspect : éducatif, pour sensibiliser les joueurs au développement durable ; ludique, pour donner envie d’y jouer. Le graphisme et le son y sont particulièrement soignés.

■ Claire Remy

L'une des caractéristiques du projet est sa pluridisciplinarité et la collaboration à distance. Maximilien Paitel, Frédéric Pedro et Anthony Chen Kuang Piao sont en première année du cycle ingénieur de Supinfo. Nicolas Gryman termine ses études cette année. Les deux premiers ont travaillé ensemble pendant deux ans à SupInfo Bordeaux, et ont toujours collaboré sur divers projets (site web, projets d'école, etc.). Frédéric et Nicolas ont fait connaissance lors de la création du laboratoire SupInfo XNA. Puis Maximilien est parti à Londres, tandis que Frédéric allait à Paris, où il a rencontré un autre étudiant de cette classe, Anthony. “ La distance entre les membres de l'équipe et le manque de temps de certains ne nous ont permis d'organiser qu'une seule réunion en face à face ! Cela aurait pu causer des problèmes de communication, mais nous les avons résolus en améliorant l'organisation : dans le projet, chacun avait son rôle et connaissait ceux des autres ”, soulignent Maximilien et Frédéric.



**Maximilien Paitel** (Supinfo Bordeaux), le graphiste de l'équipe, a commencé sa formation en autodidacte sur le web il y a deux ans. “ C'est le jeu qui m'a amené à l'informatique ”, indique-t-il. “ J'ai débuté avec les consoles Nintendo et l'Atari de mon frère. Quand les premiers ordinateurs sont

arrivés, j'ai commencé à bidouiller et à m'intéresser à l'envers du décor, et je me suis lancé dans la création. C'est surtout le côté artistique, l'infographie, qui m'attire. Je continue à faire de l'infographie en autodidacte, en parallèle avec l'école. Le jeu vidéo est une activité pluridisciplinaire avec un côté interactif pour l'interface homme-machine. Nous avons formé une équipe polyvalente. Je m'occupe surtout de la partie graphique et son, ainsi que de la traduction-localisation du jeu. ”

Quant à **Frédéric Pedro**, après 4 ans dans la presse spécialisée, il a travaillé à temps partiel comme producteur associé. Pour lui, “ Le jeu vidéo, c'est une vocation, je voulais travailler dans ce domaine, je lis la presse et les informations sur le web. J'ai travaillé chez MKO Games (filiale de MK2), j'ai développé des petits jeux vidéo (type Tetris, Pac-man...). J'utilise intensivement la plate-forme de développement de jeux vidéo XNA, ce qui permet de se concentrer sur l'idée de jeu vidéo, ainsi que SDL, outil libre dans le même esprit. J'ai créé un laboratoire commun aux différentes écoles Supinfo [SupInfo XNA, ndlr] pour partager les connaissances sur le jeu vidéo. Ça ne fait pas sérieux, mais il y a un travail monstrueux pour faire un jeu vidéo, notamment dans la relation avec l'utilisateur : comment va réagir le joueur ? Quelles émotions le jeu va-t-il susciter ?... ”

Et maintenant ? La compétition a donné

envie à l'équipe de continuer. Une société a été fondée : ECO-Think et une version arcade Xbox est en préparation. La quête d'un éditeur a démarré dans l'espoir de commercialiser le jeu.

Site : <http://www.ecological-tycoon.com>





# Votre pack internet est-il vraiment adapté ?



À PARTIR DE  
**7,80€ HT/an**  
soit 9,33 € TTC/an

**AVEC LE PACK  
À LA CARTE D'AMEN,  
VOTRE HÉBERGEMENT  
S'ADAPTE SIMPLEMENT  
À L'ÉVOLUTION DE  
VOS BESOINS.**

- Votre nom de domaine inclus
- Espace d'hébergement de 0 à 20 Go
- Php4, Php5, Perl, Python, Ruby
- De 0 à 20 bases de données avec espace de stockage flexible de 25 Mo à 1 Go
- De 0 à 200 comptes email 2 Go webmail/pop/imap
- Paramétrage de l'envoi de mails php
- Outil de création de site inclus
- Outil de création de boutique e-commerce inclus



Pour plus de renseignements : 0892 55 66 77 (0.34 €/mn) ou [www.amen.fr](http://www.amen.fr)

**NOMS DE DOMAINE - EMAIL - HÉBERGEMENT - CRÉATION DE SITE - E-COMMERCE - RÉFÉRENCIEMENT**



Questions/  
réponses avec

**Frédéric Pedro,  
ECOThink**

*Programmez : ImagineCup*

*est une course contre le temps, quelles furent les étapes avant la grande finale de Paris ?*

**Frédéric Pedro :** Le rythme de la compétition a en effet été extrêmement intense alors que nous avons commencé très tardivement ! De Mi-décembre à mi-janvier, j'ai rédigé les spécifications fonctionnelles du projet. Il nous restait seulement 1 mois et demi pour réaliser une première version du jeu et passer le round 1. Un vrai sprint ! Nous avons beaucoup de mal à nous en remettre lorsqu'un petit matin de mai, nous avons vu notre nom dans les 6 projets finalistes mondiaux...

**P :** *D'un point de vue technique, comment fut développé le projet ? Avec quels outils ? Quels furent les problèmes techniques rencontrés dans le développement ?*

**FP :** Pour remédier au problème du télétravail, nous avons privilégié le développement itératif et des solutions collaboratives comme Wiki/Trac et SVN. D'un point de vue purement technique, nous avons utilisé XNA pour le développement du jeu lui-même et .Net / C# pour les différents outils (Map Editor, l'intégrateur d'articles pour la librairie écologique, le moteur de variables, etc.). Les mécanismes du jeu étant plutôt complexes et les statistiques qui en découlent nous ont donné beaucoup de fil à retordre tant sur le plan de l'équilibrage que sur celui de la programmation de tous les cas particuliers. XNA est aussi un framework "jeune". L'absence du support natif de la vidéo et d'une GUI nous ont particulièrement ralenti.

**P :** *Développer un jeu est-il plus difficile qu'une application ?*

**FP :** Je pense que le degré d'exigence du "consommateur" n'est pas le même : soit un jeu est bon, soit il est mauvais. Il n'y a pas d'autres jugements. Les attentes sont également différentes, un jeu ne peut pas se permettre d'être austère graphiquement par exemple. Enfin, le jeu vidéo nécessite la conjugaison des talents très divers : graphistes, animateur, game designer, level designer, programmeur, statisticien, musicien compositeur, etc. Cette "alchimie" n'est pas forcément évidente car il faut parler un même langage et le "résultat" n'est visible que sur du long terme.

■ François Tonic

## Smart Cooking, la cuisine devient écologique

■ Claire Remy



### Catégorie : Software Design

L'équipe Well K'Home est éminemment pluridisciplinaire. Elle se compose d'un développeur de l'architecture, Sébastien Warin (Supinfo Lille), de deux spécialistes marketing, Gauthier Chanliau et Régis Hanol (Supinfo Montpellier), et d'un designer, Jean-Noël Gauthier (Ecole des Gobelins à Paris). Leur mentor est Laure Portet. " Dans le projet Smart Cooking, nous avons trouvé plusieurs opportunités : l'idée du gaspillage alimentaire (au niveau mondial, le quart de la nourriture produite est

duit. Les deux poubelles intelligentes sont destinées aux emballages : elles doivent apporter une assistance à l'utilisateur pour lui indiquer dans quelle poubelle les jeter.



**Régis Hanol**, étudiant à SupInfo Montpellier, chef de projet et passionné par la conception de logiciel, s'est chargé de la partie intégration du matériel RFID. Ancien participant à l'Imagine Cup, il a déjà pris part à trois finales internationales, au Japon, en Inde et en Corée, lui permettant d'acquérir l'expérience nécessaire pour mener à bien ce projet.



jetée), avec la volonté de réduire ce gaspillage pour améliorer l'environnement", explique le porte-parole de l'équipe. Cette "cuisine intelligente" comprend un réfrigérateur muni d'un écran tactile, un système d'identification RFID et deux poubelles intelligentes. L'écran du frigo affiche les données relatives à ces produits, notamment leur date limite de consommation. Lorsqu'un produit arrive à échéance, une alerte se déclenche et l'écran affiche une recette pour utiliser ce pro-



**Gauthier Chanliau** (SupInfo Montpellier) est le "Marqueteur". Etudiant en ingénierie informatique avec licence marketing, il s'est occupé de la partie "business et communication" du projet ; sa tâche est de trouver comment en tirer des revenus. Sa double compétence (marketing et informatique) lui a aussi permis d'aider au développement du logiciel. " J'ai commencé tout jeune dans l'informatique. Je voulais savoir comment sont faits les jeux. J'ai donc atterri dans le monde de l'informatique. "



Développeur principal du projet, **Sébastien Warin** (SupInfo Lille) est à l'origine de l'architecture du logiciel et de l'intégration graphique. L'animation du logiciel a été rendue possible par son travail.



Etudiant designer aux Gobelins, **Jean-Noël Gauthier** est chargé de créer l'interface. Sa spécialité est le design graphique. Designer de talent, il s'est occupé de l'aspect visuel ; il a réalisé la charte graphique ainsi que l'univers particulier du logiciel. Sa simplicité d'utilisation est traduite dans une interface claire et fonctionnelle. " Je dessine depuis tout petit, je fais des affiches, des sites internet, des brochures. De graphiste 'print', j'ai évolué en intégrant différentes technologies. La partie animation du logiciel a été faite par le développeur. La spécificité de ce projet, pour moi, a été de construire des images qui

devaient être intégrées dans le logiciel. ", explique-t-il. Ce que lui a apporté Imagine Cup : " Le développement de projet implique de s'intégrer dans une équipe pluridisciplinaire. "



Quant à **Laure Portet**, Mentor de l'équipe et initiatrice du projet, elle est étudiante en troisième année de Pharmacie. Son rôle a été d'apporter des idées nouvelles, d'organiser le travail et de fixer les objectifs. Elle a su être présente tout au long de la compétition pour donner des retours et encourager l'équipe. Comme les autres équipes, l'une des difficultés qu'ils ont éprouvée a été le temps limité pour terminer le projet : " Nous nous sommes rejoints sur Paris à partir du week-end précédant la finale. SupInfo nous a volontiers prêté une salle de cours pour que l'on puisse s'entraîner, s'entraîner et encore s'entraîner. Nous avons vraiment beaucoup travaillé, en particulier au niveau du design des slides, ainsi qu'au niveau des intonations. " site : <http://www.smartcooking.fr/>

## La partie technique



Conceptuellement abouti, le projet Smart Cooking l'est aussi sur la partie technologique. Nous avons été assez impressionnés par l'intégration des différentes couches techniques. Le projet utilise un écran tactile dont le contrôle est transparent (surcouche système du constructeur). Côté outils, l'équipe a utilisé Visual Studio 2008, SQL Server 2008 et Windows Vista (pour la partie mobile : Windows Mobile 6). Tout a été réalisé et codé en .Net 3.5 avec une utilisation intensive de WPF (et réalisation de l'interface avec Expression Blend), de WCF (pour la partie communication et services web). Pour l'intégration des lecteurs et puces RFID, l'équipe a utilisé le SDK du constructeur : TAGSYS. ■ F.T.

**Salon Les Jeudis**  
Emploi Informatique et Ingénierie

100 000 postes  
à pourvoir par an

12 salons en France  
et à l'étranger

**Vous trouverez  
forcément un emploi  
près de chez vous !**



Renseignements et pré-inscriptions: <http://www.lesjeudis.com/salons-informatiques>



## Imaginer le caddie écologique de demain : le défi de The Shads

Etudiants en première année à l'Epita, les quatre compères (3 garçons, 1 fille), The Shads (en référence à leurs pseudos commençant tous par shad : Shadok, shadodine, shadow, et shadyis-back). Leur mission : relever le défi du développement embarqué et de l'écologie avec leur projet ABC...

■ Nia Toukis

Mais pour arriver jusqu'à la finale, il a fallu de la patience, des efforts, de la sueur pour gagner le passe magique parmi les 13 000 étudiants français participant à ImagineCup. Et pour un premier essai, les Shads ont fait fort : parvenir jusqu'en demi-finale avec un caddie intelligent et écologique !

La bonne idée pour se présenter à Imagine Cup, les Shads racontent l'avoir trouvée le jour du discours de Bill Gates à la Sorbonne. Ils la tirent d'une anecdote racontée par leur coéquipier **Alexandre Reiss**, 21 ans, alias Shadisback. Le jeune homme à la silhouette élancée, chevelure bouclée blond vénitien, explique : " *Un jour chez moi, après le goûter, j'aperçois sur la table un énorme tas de déchets et d'emballages. Ce n'est pas la première fois que je le vois et que cela me choque, donc je me tourne vers mes parents puisque c'est eux qui ont fait les courses, et pour la énième fois je leur dis que ce serait mieux de choisir des produits qui ont un emballage réduit. Mais comme ce n'est pas moi qui fais les courses, j'avais des scrupules à leur faire la leçon. Je me suis dit que le mieux serait de leur apporter l'aide au moment où ils en avaient besoin, c'est-à-dire quand ils sont dans le magasin. Donc j'ai pensé à un petit système informatique qui soit un coach personnel pour faire les courses.* "

Et c'est ainsi qu'ils décident de faire équipe pour donner naissance à un chariot technologique. Avec leur caddie intelligent, la personne qui parcourt les allées d'un supermarché, qu'elle soit adulte ou même un enfant de huit ans, scanne le paquet et obtient un visuel sur l'écran placé sur le chariot : par exemple, l'emballage

est bon mais pas le transport. Le produit vient de trop loin, il a nécessité un transport longue distance coûteux en essence ou en CO2. L'écran du chariot lui affichera le verdict par un smiley ou une tête pas contente, et le consommateur fera son choix en connaissance de cause.

### Un développement intensif

Les quatre développeurs se donnent à fond dans leur projet. Ils savent que l'implication est le moteur du succès, et se sont préparés mentalement à la compétition au Centre de Formation des rugbymen. Une bonne expérience ! L'implication, confie **Sébastien Mornas**, alias Shadow 78, c'est ce qui lui avait fait défaut lorsqu'il avait tenté sa chance, seul, lors d'une précédente Imagine Cup. Le jeune homme aux cheveux mi-longs et à l'allure décontractée a découvert avec les autres membres de l'équipe la programmation pour outils embarqués. " *Tout ce que j'ai fait, c'est une calculatrice et*

*un Connect 4 sur un PocketPC fonctionnant avec Windows Mobile 5. Chacun de nous avait un téléphone mobile et un PocketPC, mais on n'avait jamais encore vraiment programmé pour ces terminaux, ça a été une expérience très excitante pour nous* ". Pour leur système embarqué, ils utilisent Windows Embedded CE 6.0, Visual Studio 2005 Professional, Microsoft Platform Builder 6.0 Add In pour Visual Studio, .NET Compact Framework 2.0, Visual C# 2.0, ICOP eBox-4300.

Un peu déçus d'avoir manqué les marches du podium, l'équipe est néanmoins déterminée à poursuivre le développement de leur chariot intelligent car ils en sont sûrs : " *On a développé quelque chose de solide, on s'est impliqués dessus, notre application peut survivre après Imagine Cup. On a envie de la voir dans les supermarchés et on veut que ce soit gratuit pour l'utilisateur, que ce soit un service fourni par les magasins* ".



Alexandre Reiss



Sébastien Mornas



### Une fille experte en développement...

La seule fille de l'équipe, Caroline Vigouroux : 20 ans, alias Shadoline, apparaît plutôt timide. Mais attention, c'est un cerveau aiguisé !

" *J'adore les maths. L'informatique est un outil qui me permettait de mettre en application les maths. Je ne me souviens plus très bien quand j'ai décidé de faire de l'informatique, mais c'était dès l'école primaire. En fait, je trouvais cela assez fascinant. Chez moi, on avait une grosse machine, un système avec Windows 3.1, et je me souviens qu'on ne pouvait pas faire grand chose. Il fallait passer par la console pour aller faire des jeux. Il y avait un informaticien qui m'aimait bien, et qui s'amusait à cacher des jeux dans la machine et pour les trouver, j'étais obligée de parcourir tous les répertoires pour aller chercher le jeu en question. Un vrai jeu de pistes qui m'a donné le goût de l'informatique. Cela m'a toujours plu d'aller à tâtons, et de découvrir au fur et à mesure l'informatique. Je tenais absolument à devenir ingénieur, et on m'avait conseillé de passer d'abord par maths sup et maths spé. Donc j'ai fait maths sup, mais je voulais vraiment faire de l'informatique, pas être ingénieur généraliste, être ingénieur en informatique, alors j'ai rejoint l'EPITA après un an de maths sup.* "



## RESEAU :

# Médaille d'or pour Jean-Benoît Paux

Les équipes françaises se sont distinguées durant toute la durée de la finale. Le challenge IT, centré sur le réseau et le serveur, est difficile, car se déroulant en solitaire durant 24 heures non stop ! Mais, la récompense était là... une coupe, et un chèque de 8 000 dollars. Belle récompense après une 4e place dans l'édition 2007 se déroulant à Séoul.

■ Propos recueillis par Nia Toukis

### *Programmez : D'où vous est venue votre passion pour l'informatique ?*

Jean-Benoît Paux : J'ai découvert l'informatique assez jeune, à l'âge de 6 ou 7 ans. D'abord par les jeux et les programmes éducatifs. Mes parents ne m'ont ni poussé spécialement vers l'informatique, ni limité ni interdit d'y toucher. L'informatique n'est pas leur domaine: mon père est médecin et ma mère femme au foyer. Mon père s'était acheté un ordinateur, un Macintosh, lorsque j'avais 6 ans et il m'a laissé m'en servir. Au début, c'est lui qui m'a fait découvrir l'ordinateur et puis après, l'élève a dépassé le maître ! Je me suis intéressé tout seul au fonctionnement interne de la machine, je voulais voir comment ça marchait. Par la suite, j'ai approfondi avec des copains, parce qu'à plusieurs, on partage, et le reste est venu de fil en aiguille.

### *P ! : Pendant la compétition de Imagine Cup 2008, en quoi consistaient les épreuves de IT Challenge ?*

JBP : Pour les premières épreuves, on est plusieurs milliers. Tout le monde passe un QCM à la même heure sur internet sur des questions posées par Microsoft. Après cette épreuve de qualification, on est environ 500 à 1000 pour le deuxième round. Là, un scénario nous est fourni : une entreprise souhaite se développer et nous demande de concevoir sur papier tout son système informatique de A à Z. C'est un peu comme si on devait répondre à un appel d'offres et proposer la meilleure solution. On a une vingtaine de pages en anglais avec des schémas et des tableaux et on doit décrire quels produits on utiliserait, comment on les agencerait, le nombre de serveurs, le dimensionnement, comment on générerait la sécurité, les sauvegardes, etc.

### *P ! C'était avant les épreuves de Paris ?*

JBP : Tout à fait. Cela se déroule en avril -

mai. Cette épreuve décide si on est qualifié pour la compétition mondiale, la finale, se tenant à Paris.

### *P ! Et l'ultime épreuve se déroule durant la grande finale ?*

JBP : Oui. On nous redonne à peu près le même scénario. Nous avons alors 24 heures pour installer, configurer l'ensemble des points du système. On a débuté le vendredi à 24h pour terminer le samedi à 14 heures.

### *P ! : Qu'est-ce qui était le plus dur ?*

JBP : Sans doute d'arriver à gérer mentalement quand les problèmes surviennent, de résister sans perdre espoir. Et aussi, de rester 24 heures devant l'écran, car il faut lutter pour rester éveillé.

L'année dernière, j'étais complètement épuisé durant l'épreuve et j'avais du mal à résister au sommeil.

Cette année, j'avais pris mes précautions pour éviter ce problème : j'ai bu pas mal de café et des boissons énergisantes, ce qui m'a donné d'ailleurs un peu de mal à m'endormir par la suite ! Au moins ça m'a permis de tenir jusqu'au bout des épreuves. A la fin, j'en avais quand même assez de rester devant l'ordinateur. En 24 heures, on a monté une maquette de système informatique, cela correspond à ce qu'on fera plus tard professionnellement.

### *P ! : Et les pièges du jury durant l'épreuve ?*

JBP : On doit tout configurer correctement, et à un moment, on vous prévient que pendant une heure, vous devez être prêts pour une attaque.

Les juges vont essayer de tester la résistance de votre système, par exemple faire tomber un serveur et vous devez montrer que votre système fonctionne quand même. Ils vont sniffer les communications pour voir si les données sont bien encryptées.

### *Comment avez-vous traité la thématique de l'environnement ?*

JBP : Ce qu'on appelle le Green IT commence à bien se développer : c'est très intéressant, cela concerne par exemple tout ce qui est consommation électrique. Il y a plein de solutions pour diminuer la consommation, par machine ou par watt. On l'a traité surtout dans le round 2, la deuxième série d'épreuves de qualifications d'Imagine Cup. Je crois que cela m'a apporté quelques points. Dans le round final, il s'agissait de mettre en pratique un certain nombre de recommandations.

### *Vous n'aviez pas trop le stress devant le jury ?*

JBP : A l'école on a pas mal de projets, de soutenances, donc on commence à être rôdé. Ici, la particularité est que tout se déroule en anglais. Mais nous savons à peu près ce qu'ils peuvent demander. Il faut essayer de montrer ce qui marche et éviter de parler de ce qui marche moins bien...

### *Participer à Imagine Cup et en sortir vainqueur vous a ouvert des portes ?*

JBP : Oui ! Des gens m'ont demandé de les contacter après les vacances, pour la plupart ce sont des personnes qui travaillent dans des sociétés de services.

### *Quels conseils pourriez-vous donner aux futurs candidats ?*

JBP : Etre passionné, c'est le plus important ! Et savoir utiliser les logiciels. Ce n'est pas facile de maîtriser une vingtaine de logiciels Microsoft. Pour les épreuves de IT Challenge, j'ai utilisé Windows Vista, Windows Serveur 2008, Exchange, SQL Serveur, la suite System Center, etc. L'INSA a un accord avec les Ministères pour faire le support Microsoft de l'ensemble des Universités, et dans ce cadre là, j'ai pu me familiariser à certains produits de la bibliothèque Microsoft.

# And the winner is... l'équipe australienne !

## Projet SOAK : économiser 50% de l'eau du pays !

Le suspense est à son comble. La grande salle du carrousel du Louvre est pleine à craquer. Le moment d'annoncer le grand vainqueur de l'édition 2008 d'ImagineCup arrive enfin... Le pays des kangourous, l'Australie, est récompensé pour le projet SOAK, de l'équipe éponyme. L'équipe a montré un projet solide, parfaitement monté et développé, et surtout, opérationnel dans plusieurs fermes du pays.

■ Nia Toukis



L'équipe et sa mascotte.

L'équipe se compose de quatre étudiants. L'idée de départ était de développer une application permettant de gérer l'utilisation de l'eau dans les fermes pour faire face à la sécheresse. Pourquoi SOAK comme nom ? SOAK signifie Smart Operational Agricultural toolKit. Et l'équipe avait fortement impressionné le jury et une grande partie de la presse dont Programmez !.

### Une application écologique pratique et réelle

Les deux tiers de l'eau, en Australie, continent sec, sont utilisés pour l'agriculture. L'enjeu est de taille ! Or nos étudiants estiment qu'ils pourraient faire économiser 50% de cette eau, en optimisant l'arrosage. Soit la moitié de l'eau consommée en Australie ! David Burela, la tête pensante du projet, cheveux bruns ébouriffés et regard pétillant, explique que l'idée de ce logiciel leur est venue parce que le problème de l'eau est crucial en Australie. Les quatre programmeurs pensaient d'abord développer un logiciel pour économiser la consommation d'eau chez les particuliers, mais ils se sont vite rendus compte qu'il y avait



Carte des champs

un problème beaucoup plus grave : le manque d'eau pour les agriculteurs. " Ce qu'on apprenait était effrayant, raconte David, ce manque d'eau pousse les fermiers au désespoir : tous les quatre jours, l'un d'entre eux se suicide parce qu'il n'a pas assez d'eau pour arroser ses cultures ou abreuver son bétail. Il est si difficile pour eux de continuer à vivre de leur exploitation à cause de ce grave problème. Voilà pourquoi nous avons voulu développer une solution qui les aide à le surmonter".

" Seulement voilà, poursuit son coéquipier Ed Hooper, on n'y connaissait rien au travail dans une ferme ! ".

" Nous ne sommes pas des paysans, nous ne pensions pas comme eux ", commente David. Les voilà donc partis se renseigner sur le terrain en questionnant des fermiers et en visitant une véritable exploitation.

Avant de commencer à coder la moindre ligne de leur programme, ils consacreront deux mois et demi à étudier scrupuleusement le fonctionnement d'une ferme. Celle qui leur a servi de modèle, baptisée par son propriétaire " Morning Star " (Etoile du matin) est idéale parce qu'elle a une activité variée : un élevage de vaches et de moutons, ainsi qu'un vignoble et des cultures de roses.

### Des capteurs à 60 euros, au lieu de 1000 !

Les étudiants découvrent que les cultures sont arrosées de même façon uniforme. Pourquoi ne pas arroser les sols de façon plus ciblée en fonction de leur nature, du type de culture qu'ils accueillent et pourquoi mettre en marche les arroseurs alors qu'... une pluie est annoncée par les prévisions météo ?

Après des semaines d'interviews avec des agriculteurs, des jours passés



dans les bibliothèques et dans les champs, ils savent quelles fonctionnalités seront les plus utiles. Voilà le principe du développement de leur logiciel trouvé. Le programme SOAK reçoit ses informations par des senseurs placées sur les différentes parcelles de l'exploitation agricole. Les senseurs équipés de panneaux solaires envoient les informations météo par radio au système qui dira à l'agriculteur quelle parcelle est la plus sèche en ce moment et a besoin d'être arrosée. Les infos météorologiques sont également entrées pour éviter un arrosage si de la pluie s'annonce dans les heures à venir, ou si le sol de l'une d'entre elles est déjà détrempé. " Avec notre application, le fermier peut gérer ses réserves d'eau et éviter de les gaspiller ". L'application lui donne toutes ces informations sur son PC, mais il peut également les consulter en plein champ sur un PDA. Les informations sont présentées sous forme de graphiques et de cartes. L'aspect visuel a été particulièrement travaillé par l'équipe de programmeurs. Le logiciel gère également les arroseurs de l'exploitation agricole, en les coupant ou en les mettant en marche au meilleur moment.

### Etudiant et déjà consultant

Ces quatre développeurs, venant de 4 universités différentes, se sont rencontrés lors d'un TechEd en Australie et ont décidé de faire équipe pour développer cette solution. Pas toujours facile de se retrouver lorsque chacun habite un état différent en Australie. David Burela, vit en Tasmanie, territoire situé au large du Sud Est de l'Australie. Il a choisi de suivre une double formation universitaire, à l'université d'Hobart et a obtenu un Master en Informatique, ainsi qu'un MBA. Cette connaissance du monde des affaires n'est sans doute pas pour rien dans l'aisance et le pouvoir de conviction que dégage ce jeune homme de 24 ans lorsqu'il mène la démonstration de SOAK devant les jurys. A côté de ses études, David travaille actuellement comme Senior Developer, consultant et formateur

## Un festival de nouvelles technologies

Pour créer les " zones " (les champs) du fermier concernés par SOAK, l'application utilise Live Maps pour avoir une vue aérienne de l'exploitation. On définit les zones concernées, les données des niveaux de l'eau étant liées à une zone précise. Les données, stockées dans une base SQL Server, se divisent en plusieurs catégories : niveau de la nappe, précipitations, vent, température, débit de l'eau, niveau d'humidité. Ces données sont fournies par des capteurs. La base se divise en sous-bases pour chaque type de données. A cela se rajoutent les sources d'eau (eau de pluie, eau de sources et l'eau recyclée). Le front-end de l'application, pour l'affichage et la gestion, est disponible sur 3 plateformes : Web (administration, contrôle et analyses), PDA (surveillance des niveaux, information par zone, affichage des avertissements) et enfin un widget Vista pour les niveaux d'eau et l'environnement de la ferme.

L'architecture SOAK est multi-tiers et facilement adaptable grâce à des web services et à la séparation des couches. Toutes les nouvelles technologies .net ont été utilisées comme Silverlight 2, Linq pour le requêtage des données, Windows Live pour la cartographie, WCF pour la partie web service, XML.

■ F.T.



Un des capteurs de SOAK



L'application sur mobile



chez Readify, 60 employés, une des deux seules sociétés australiennes spécialisée en .Net. Il est fier de son salaire : 90 000 dollars australiens, le double du salaire d'un ingénieur débutant.

Ses trois autres co-équipiers habitent dans l'Etat de Victoria. Ed Hooper, 23 ans, chevelure blonde, carrure imposante, étudie l'informatique à l'Université de Melbourne. Le discret Long Zheng, 20 ans, a choisi un cursus d'affaires et économie, ainsi que d'informatique à l'Université de Monash et Dimaz Pramudya, 20 ans, visage toujours souriant, suit des études d'ingénieur informatique à l'université

de Technologie de Swinburne. " On communique par emails, par téléphone et parfois même par fax ! " plaisante Dimaz.

Leur projet les a passionnés et ils prévoient de le commercialiser avec une idée en tête : que le produit ne coûte pas trop cher aux agriculteurs. " Il faut trouver le bon équilibre entre le coût, et la facilité d'emploi pour les agriculteurs, sinon, ils ne s'en serviront pas ", explique David Burela. " J'aimerais que les senseurs à panneaux solaires ne reviennent pas plus cher qu'une soixantaine d'euros, alors que les capteurs du marché, à eux seuls, coûtent de 1000 à 2000 euros. Ainsi les agriculteurs n'auraient pas à dépenser, en totalité, plus de deux ou trois mille euros en moyenne pour maîtriser leurs réserves d'eau avec notre système. Un système actuel coûte souvent 100 000 euros ou plus ".

Après la compétition, et forts de leur victoire, les auteurs de SOAK pensent à la commercialisation de leur solution et sont déjà en pourparlers pour obtenir le soutien du Parti National Australien et celui des fermiers australiens. A Programmez, nous parions que le projet est une graine de multinationale !

# L'INFORMATION du DÉCIDEUR

## Choisir, déployer, exploiter les logiciels

Abonnez-vous au seul magazine offrant aux **responsables informatiques** une information et des témoignages focalisés sur le logiciel en entreprise.

**Dans chaque numéro,** les tendances, les dossiers, les interviews, les témoignages, les avis d'expert dans tous les domaines du logiciel professionnel :

- Les SSII, des métiers et du recrutement ;
  - L'administration, les réseaux ;
  - La sécurité, la sauvegarde ;
- La gestion des projets, les méthodes, le développement ;
- Les progiciels, ERP, BI et SGBD...



### L'actualité au quotidien :

- Sécurité • Projets et développement
- Administration
- Progiciels

### Les Cas Clients

Prochainement : **Vidéos** (Actualité et Cas Clients)

[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 : 25€ au lieu de 30€, prix au numéro** (Tarif France métropolitaine) - Autres destinations : CEE et Suisse : 30€ - Algérie, Maroc, Tunisie : 33€ , Canada : 39,50€ - Dom : 38€ Tom : 50€  
6 numéros. Prochaines parutions : N°4 Octobre - N°5 Novembre/Décembre - N°6 Janvier/Février 2009 - N°7 Mars/Avril - N°8 Mai/juin - N°9 Juillet/Août/Septembre

☐ 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



# MICROSOFT... l'après Bill Gates

Où va Microsoft ? Qui dirigera demain la société ? Mary Jo Foley, journaliste et blogueuse indépendante révèle dans son livre « non autorisé » ce que sera « Microsoft 2.0 ». Programmez vous offre dans ce numéro le chapitre d'introduction du livre à paraître le 24 septembre chez Dunod.

*Bill Gates est en quelque sorte le pape de notre industrie et, en tant que pape, la foule est suspendue à ses lèvres. Si vous voulez aller au paradis, il vous faut la bénédiction de Microsoft.*

Gary Shapiro, DG de Consumer Electronics Association, *The Seattle Times*, 31 décembre 2007

## Microsoft 2.0 : bienvenue dans la nouvelle entreprise de l'ère post-Gates

« À 5 heures, le 1er juillet 2008, l'alarme de son Zune réveille Steve Ballmer, le PDG de Microsoft, et le projette dans une nouvelle réalité.

C'est le premier jour où le Président Bill Gates n'est plus officiellement aux commandes de Microsoft. Comme il l'a annoncé il y a maintenant deux ans, Gates a décidé d'abandonner ses responsabilités opérationnelles pour se concentrer sur sa Fondation. Gates reste toujours Président de Microsoft, mais il n'interviendra plus dans le suivi des logiciels, les conférences et la prise de décision quotidienne.

« Encore une fois, Bill retire toute la gloire, et je me retrouve dans le pétrin », soupira Ballmer en repensant aux chiffres de vente de Halo 3. « Les clients et nos partenaires sont en guerre contre Vista. Et puis il y a ces allumés de la sphère Linux et leur mépris des brevets. Alors ne me parlez pas de Google car ce n'est pas une chaise que vous allez recevoir sur le coin de la figure, mais mon fauteuil ! »

Habituellement plutôt grandiloquent, Ballmer n'en menait pas large. Microsoft commençait à partir à la dérive et il ne savait pas vraiment comment redresser la barre. Le logiciel, vendu par abonnement en tant que service était en train de supplanter à un rythme de plus en plus rapide l'industrie classique, avec ses marges confortables, du logiciel vendu dans des boîtes.

Apple, que Microsoft avait renfloué il n'y a pas si longtemps, était devenu la coqueluche des médias et de la jeune génération. Et à cause de Neelie Kroes, Colleen Kollar-Kotelly, et tous ces inquisiteurs antitrust, Microsoft ne pouvait rajouter la moindre brouille à Windows sans se retrouver devant un tribunal ! » Coupez !

C'est de cette manière que certains de mes collègues souhaiteraient que je commence mon livre sur l'avenir du nouveau Microsoft de l'ère post-Gates. Ils veulent me

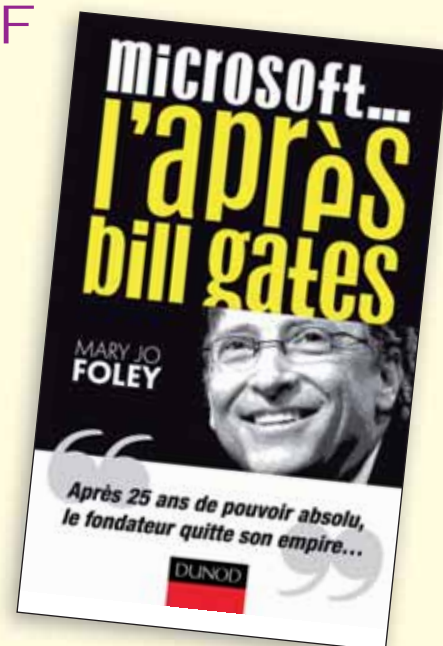
faire dire que Microsoft a perdu son empire et ne sera bientôt plus qu'une quantité négligeable de l'industrie des nouvelles technologies. Et pour ce faire, ils aimeraient que j'écrive dans un banal style réaliste, truffé de propos rapportés de la bouche d'employés de Microsoft.

Mais je ne ferai pas. Non seulement, je trouve ce style d'écriture maniéré et vulgaire, mais surtout ce n'est pas la manière dont je vois l'évolution des choses.

Oui, Microsoft est à la croisée des chemins, mais pour moi un carrefour ne signifie pas la mort. Je sais que tout le monde n'est pas d'accord. Les sociétés importantes de Wall Street prétendent que la vieille société Microsoft n'est plus ce qu'elle était et nombreux sont ceux qui prédisent que Microsoft va devoir investir son immense trésorerie pour concurrencer Google et les autres sociétés du Web 2.0. Les experts en technologie affirment que Microsoft s'est égaré et s'est transformé en une société dont les systèmes d'exploitation ne sont appréciés que par les « grand-mères » (ce sont les mots du programmeur Paul Graham, par les miens) (1).

Jusqu'à présent, Microsoft était incarné par Bill Gates. C'est lui qui a tenu tête aux avocats du Ministère de la justice des États-Unis et c'est lui qui a insisté dans sa déposition sur le fait qu'il ne connaissait pas la signification du mot « nous ». Lui, le célèbre PDG redouté par ses employés pendant les phases de test du code, et l'étudiant qui abandonna la fac pour devenir l'homme le plus riche du monde et l'un des leaders de l'informatique les plus puissants et les plus craints.

1. À la fin des années 1990, c'était Bill Gates contre David Boies. Dans le procès antitrust qui opposait le Ministère de la justice à Microsoft, Boies, qui était l'avocat représentant l'accusation, a dirigé des débats de plusieurs heures impliquant le président de Microsoft. Écoutant ses avocats qui lui conseillaient de rester sur ses gardes, Gates eut sans cesse recours à des faux-fuyants qui restèrent tout ce que l'on a retenu de son interrogatoire. Businessweek, comme d'autres médias, ont relaté la contre-performance de Gates ici : [www.businessweek.com/1998/48/b3606125.htm](http://www.businessweek.com/1998/48/b3606125.htm)





L'empreinte de Gates est partout chez Microsoft, de la manière dont les cadres s'habillent, se coiffent, se balancent sur leur chaise, jusqu'à la liste des « amis de Bill » qui traditionnellement faisaient la loi à Redmond (localité du siège de Microsoft).

Quand Gates a fondé Microsoft, Internet était très loin d'être aussi répandu qu'aujourd'hui. Google n'existait pas encore. L'industrie des logiciels pour PC était en grande partie une affaire de produits packagés (et pas de services), et personne n'avait entendu parler d'Ajax ou d'open source.

Avec l'apparition de ces nouveaux concurrents, de nombreux observateurs du monde industriel ont eu le sentiment que Microsoft était devenu encore plus lent que d'habitude à comprendre la situation et à réagir à ces nouvelles tendances, et ont dépeint Microsoft comme un has-been qui serait incapable de retrouver son rang parmi l'élite, même s'il était prêt à acheter une grande locomotive du Web 2.0 comme Yahoo. D'autres au contraire continuent à penser que Microsoft ne pourra jamais être délogé de son perchoir tout simplement en raison de sa taille et du monopole dont il a si longtemps bénéficié.

Je me situe quelque part entre ces deux extrêmes. Je sais qu'il suffit souvent d'une rude concurrence pour galvaniser le géant du logiciel. Je sais que Microsoft ne fait pas semblant de courir après Google et son « Ne faites pas le mal ». Après tout, Microsoft n'a pas atteint la position industrielle qui est la sienne depuis une vingtaine d'années en étant gentil et équitable. Le credo de Microsoft est « Tout est juste en amour et en affaires ». Il se pourrait même que sa devise soit plutôt « Même si c'est injuste, si vous pouvez vous en sortir et faire des bénéfices, foncez ! ».

J'ai passé 25 ans de ma vie de journaliste à observer et à décrire Microsoft, et je n'ai jamais été intéressée par la rédaction d'un ouvrage sur cette société car je ne fais pas partie de ceux qui passent leur temps à faire des bilans. Et puis qui écrit des livres économiques sur l'avenir ?

Et pourtant, c'est justement ce que j'ai tenté de faire avec le livre que vous avez dans les mains.

C'est un livre sur l'avenir de Microsoft qui va devenir imprévisible après le rachat de Yahoo qui interviendra certainement cette année. Dans un futur proche, une bonne partie des cadres qui dirigent Microsoft vont sans doute prendre le large, particulièrement si le cours de l'action stagne comme c'est le cas ces dernières années. Pour autant, la technologie va continuer sa course folle. Microsoft va passer des accords d'une envergure jamais envisagée auparavant afin de ne pas rester à la traîne. La peur que le géant de Redmond inspirait traditionnellement sera remplacée par la croyance stupide (c'est tout du moins mon opinion) que Microsoft va péricliter.

Je ne prétends pas posséder une boule de cristal qui me permette de prédire sans faille ce que Microsoft envisage de faire dans les prochaines années... ni même dans les prochains mois. Cependant, en me basant sur les contacts que j'ai régulièrement avec les cadres, les partenaires, les clients et les concurrents de Microsoft, je pense être en mesure de faire quelques hypothèses valides.

Cet ouvrage décrit les gens de Microsoft, les produits et les stratégies qui à mon avis seront essentiels pour la prochaine génération. Je me rends compte que ma tentative de décrire l'avenir des unités commerciales de la société est un pari impossible car ce n'est pas parce que l'on connaît bien Windows et Office que l'on est capable de discerner la manière dont les 80 000 employés de Microsoft s'en sortent.

Cela étant dit, il y a quand même de bons indices sur ce que va faire Microsoft à l'avenir. En rassemblant toutes les pièces du puzzle, on a une bonne vision de ce que sera Microsoft 2.0 dans les dix prochaines années. J'espère que mon travail d'investigation aidera les clients, les partenaires et les observateurs de Microsoft à prendre de bonnes décisions en 2008 et au-delà.

### L'avenir de Microsoft n'est pas tout rose

Avant de refermer ce livre, en le considérant comme une apologie de la part d'un thuriféraire de Microsoft (ce que vous ferez certainement), laissez-moi vous expliquer pourquoi je pense que Microsoft doit faire face à des défis redoutables.

- Ray Ozzie ne sera pas le même CSA (Chief Software Architect, Architecte logiciel en chef) que Gates. Même si Ozzie conserve le titre de CSA dont il a hérité en 2006, beaucoup de gens doutent qu'il voudra ou pourra être à la tête de la stratégie de la technologie de Microsoft. Microsoft va se démenier afin de trouver une nouvelle personnalité publique pour incarner le rôle de technologue en chef. Et il y aura de nombreuses rivalités internes au sein de Microsoft parmi ceux qui s'estiment être le mieux placé pour endosser ce rôle.
- La bureaucratie de l'équipe Windows qui est la cause du retard de Vista, système inachevé et mal accueilli par le public, est une véritable menace pour l'équipe Windows Live. Si les choses ne changent pas, les services Windows Live vont se perdre dans les processus et les procédures qui ont handicapé l'unité Windows Client. Cette rigidité finira par empêcher Microsoft de tenter de concurrencer Google, Salesforce.com et d'autres services plus agiles.
- Microsoft va continuer à protéger son produit le plus important, Windows, dans l'espoir que la bulle du Web 2.0 va éclater avant de devoir abandonner le couple

logiciel + services au seul profit des services. Il a fallu des années à Microsoft pour se décider à faire un petit pas vers un nouveau modèle de distribution des logiciels ou pour assouplir l'accès au code de ses produits.

- Microsoft s'est investi dans plusieurs domaines stratégiques des nouvelles technologies : téléphonie mobile, recherche sur le Web, publicité en ligne, et fourniture de musique et de vidéo en ligne, pour ne citer que quelques exemples. Presque chaque fois que Microsoft a sorti une « innovation » (on peut penser au Tablet PC, aux montres SPOT, aux écrans tactiles « Mira »), cela n'a débouché sur rien. Microsoft cherche des moyens d'institutionnaliser l'innovation, avec des initiatives comme les Quests, Live Labs, Office Labs, et adLabs, mais sans aucun résultat à ce jour.

Jusqu'à présent, Microsoft a profité de la puissance de son monopole, mais un Microsoft moins marqué par l'empreinte de Gates sera une entreprise sans boussole à court terme. L'équipe actuelle des dirigeants est trop imprégnée des valeurs de la vieille école pour changer de cap rapidement. Heureusement pour Microsoft, les mines d'or que représentent Windows et Office peuvent continuer à faire avancer le navire pendant encore deux ou trois ans. Mais à moins que Microsoft n'apporte du sang neuf qui ne soit pas à la recherche du statu quo, les employés vont vivre des temps difficiles à repousser les nombreux concurrents qui apparaissent.

Cela étant dit, je suis moins pessimiste sur les chances de Microsoft de rester une puissance majeure du monde de la technologie que de nombreux observateurs de cette industrie (et cela ne tient pas à ma position de journaliste à temps plein, payée pour étudier Microsoft. Si besoin est, je peux toujours changer d'orientation et devenir Madame Facebook ou Madame Google).

Pourquoi est-ce que je pense qu'il ne faut pas enterrer Microsoft ?

- Microsoft a eu en 2007 un chiffre d'affaires de 52 milliards de dollars et emploie 80 000 salariés. Microsoft envisage d'acquérir une vingtaine d'entreprises par an pendant les cinq années à venir. Des centres de traitement de données pour l'hébergement Web, composés de milliers de serveurs et valant plusieurs millions de dollars sont en cours de construction ou déjà opérationnels à Chicago, Dublin, Quincy, San Antonio, Santa Clara, Siberia et dans d'autres villes. Ce qui est rassurant, c'est que Microsoft n'est pas une de ces fumeuses sociétés Web 2.0 qui viennent d'éclore.
- La décision de Microsoft de parier sur les logiciels et les services, et non pas simplement sur les services, est une sage décision bien qu'elle ait été largement critiquée.

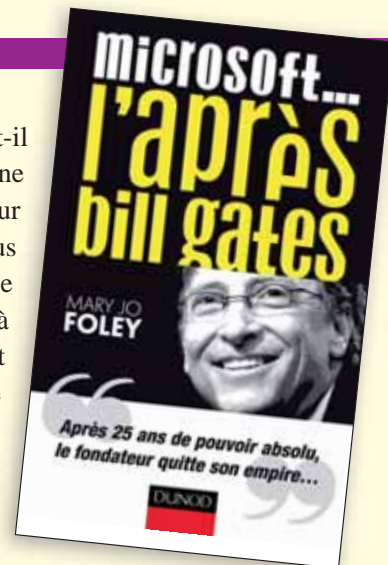
Réfléchissez : vous arrive-t-il souvent de vouloir créer une feuille de calcul à l'intérieur d'un navigateur ? Allez-vous créer beaucoup de présentations PowerPoint à partir d'un kiosque Internet plutôt que sur votre ordinateur ? (2) Tant que les bureaux virtuels ne seront pas plus fiables, la plupart des travailleurs de l'information ne sont pas prêts à parier sur cette technologie.

- Un écosystème important, rentable et étonnamment fidèle à Microsoft continue de prospérer. En dépit des procès antitrust, des actions collectives en justice et des messages hostiles à Microsoft sur Twitter, Microsoft a encore beaucoup d'adeptes. Microsoft a permis à de nombreux consultants, revendeurs et intégrateurs système de gagner très correctement leur vie. Même si Microsoft continue à s'éloigner des marchés de niche qui ont fait la richesse de ses partenaires, de nombreuses personnes créent des entreprises en relation avec la firme de Redmond.
- Cela va prendre beaucoup de temps pour détrôner Microsoft. Par exemple, malgré toutes ses erreurs commises dans la guerre des navigateurs Web, Microsoft possède encore 75 % de parts de marché. C'est la même chose avec Windows qui est toujours installé sur plus de 90 % des PC. Il est vrai qu'Apple a gagné entre 3 et 7 % de parts de marché et a une meilleure cote de popularité que Microsoft chez les jeunes (ainsi que chez de nombreux journalistes, même si ce n'est pas mon cas). Mais Microsoft a le reste et se taille la part du lion sur les marchés les plus lucratifs. Trop d'experts, de financiers, de blogueurs et de passionnés ont été un peu rapides à enterrer Microsoft. Les gens aiment bien les belles histoires, surtout quand David terrasse Goliath. Mais en réalité, même si Microsoft se fourvoie complètement dans ses décisions stratégiques et dans sa politique commerciale, il faudra encore plusieurs années pour que le géant morde la poussière.

(.../...)

---

2. Les experts continuent à déplorer la décision de Microsoft de ne pas sortir une version Web de sa suite Office. Mais la raison qui pousserait Microsoft à une telle migration est rarement étudiée. Non seulement une version Web de Microsoft Office diminuerait grandement les bénéfices de Microsoft, mais cela ne concernerait aussi qu'une frange relativement faible de la population. Vous trouverez plus d'informations sur ce sujet sur mon blog « All About Microsoft » <http://blogs.zdnet.com/microsoft/?p=1030>



## Grand public ou professionnels ?

Ozzie a déclaré qu'il croyait que les compagnies technologiques avaient besoin d'un fort segment grand public pour rester dans la course, car c'est dans le marché grand public que les innovations ont lieu aujourd'hui (3). À la fin de l'année 2007, il y a eu un débat dans la blogosphère pour savoir si les technologies grand public sont plus attirantes que les technologies d'entreprise (4) et une majorité pensent que c'est le cas.

Pour ce qui me concerne, je crois que les vendeurs se sont rendu compte que la presse parlait plus facilement des produits grand public que des produits professionnels. Les produits grand public sont faciles à comprendre et à expliquer et font souvent la une des journaux, des magazines, et des sites Web. Est-ce que le Wall Street Journal peut faire un article sur le nouveau Microsoft Zune ? La réponse est oui. Est-ce qu'il peut faire un article sur le logiciel Microsoft Exchange Server ? La réponse est non.

En réalité, Microsoft tire toujours la majorité de son bénéfice des ventes des logiciels professionnels et il ne semble pas que cette situation puisse changer dans les années à venir. Microsoft continue à injecter de l'argent dans de nouveaux secteurs grand public dans l'espoir de diversifier ses activités, ainsi que pour trouver de nouveaux moyens de rentrer dans le foyer de ses clients professionnels. L'idée est que si un utilisateur apprécie la Xbox et le Zune, il est probable qu'il envisagera aussi d'acheter des produits et des technologies Microsoft pour sa maison.

Début 2008, Microsoft a commencé à intensifier ses campagnes publicitaires et a affecté 300 millions de dollars pour une nouvelle annonce en direction du grand public. Dans les mois et les années à venir, il faut s'attendre à voir investir encore plus d'argent dans les stratégies commerciales grand public.

Pourtant, selon moi, la seule réelle « innovation » qui a lieu dans le secteur grand public concerne l'interface utilisateur ou, comme Microsoft préfère l'appeler, « l'expérience utilisateur ». Rendre les logiciels plus faciles à utiliser, à mettre à jour, à déployer et à maintenir est la nouvelle quête du Graal, pas seulement pour Microsoft, mais pour tous les éditeurs de logiciels. Les utilisateurs veulent des PC et des téléphones mobiles aussi faciles à utiliser que des téléviseurs et des baladeurs numériques. En fait, l'expérience utilisateur se situe là où les technologies grand public et les éditeurs nous mènent, c'est-à-dire vers le monde professionnel. C'est aussi pour cette raison que je

me concentre plus sur les stratégies et les produits professionnels de Microsoft que sur le marché grand public, que ce soit sur mon blog ou dans ce livre.

### À propos de l'expression « Microsoft 2.0 »

J'ai des sentiments partagés sur l'utilisation de l'appellation « 2.0 » pour décrire l'avenir de Microsoft.

Tout d'abord, Web 2.0 est l'un des termes les plus galvaudés et les plus incompréhensibles du marché des nouvelles technologies. C'est aussi un terme connoté qui accrédite la thèse que le Web 2.0 est le remède universel pour tout ce qui cloche dans le domaine technologique. Beaucoup de gens emploient le terme Web 2.0 pour différencier les sociétés, les idées et les technologies que ne partagent pas les locomotives actuelles du secteur informatique que sont Microsoft, IBM, SAP, Oracle, etc.

Mais au cours des mois qui viennent de s'écouler, les dirigeants de Microsoft ont finalement commencé à intégrer la marque et les concepts du Web 2.0. De plus, Ozzie croit apparemment que le Web 2.0 est l'avenir. Microsoft, qui a par le passé tourné en ridicule le Web 2.0, est devenu un de ses plus grands partisans. Le nombre grandissant des services de la famille Windows Live, avec leurs cartes, leurs blogs, leurs flux RSS, et leurs réseaux sociaux, sont dans toute leur splendeur des exemples d'applications Web 2.0 pour le meilleur et pour le pire. D'autres équipes Microsoft trouvent le moyen de renommer et de repositionner leurs produits pour les présenter comme des mashup, des mondes virtuels et des contenus générés par l'utilisateur.

Si Microsoft veut commencer à parler de Web 2.0 (mieux vaut tard que jamais), je trouve qu'il est équitable d'employer le terme « Microsoft 2.0 » pour décrire le futur de la stratégie, des produits et du personnel de cette société. Microsoft 2.0 est donc la formule que j'emploie pour désigner l'avenir de Microsoft. Microsoft 2.0 implique un nouveau Microsoft, centré sur le Web 2.0. Cela englobe aussi tous les autres produits, services, stratégies et technologies qui ne font pas partie du Web 2.0 et qui vont guider Microsoft au cours des prochaines décennies.

Mary Jo FOLEY,  
Extrait de : **Microsoft... l'après Bill Gates**,  
Avec l'aimable autorisation des éditions Dunod.  
Parution 24 Septembre 2008.

**Un nouvel extrait, le mois prochain : Microsoft 2.0**

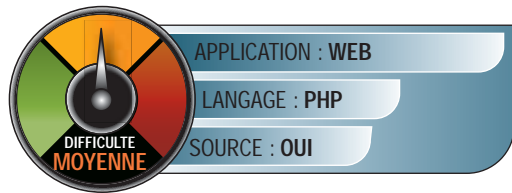
3. Ray Ozzie, le CSA de Microsoft a déclaré officiellement en 2007 que les plus grandes innovations proviennent aujourd'hui du marché grand public plutôt que du secteur professionnel. Vous trouverez une transcription de ses propos ici : [www.microsoft.com/msft/speech/FY07/QAFAM2007\\_2.mspx](http://www.microsoft.com/msft/speech/FY07/QAFAM2007_2.mspx)

4. Robert Scoble, ancien employé de Microsoft, a entamé ce débat en décembre 2007 en postant sur son blog un article intitulé « Pourquoi les logiciels professionnels ne sont pas attirants ? » : <http://scobleizer.com/2007/12/09/why-enterprise-software-isnt-sexy>



# Un moteur de templates en PHP 5

Tout webmaster ayant déjà programmé un site professionnel a dû se confronter à la difficulté du multi-navigateur. De grosses différences continuent d'exister entre divers concurrents : un site peut très bien s'afficher sous Mozilla Firefox et ne ressembler à rien sous Internet Explorer (ou inversement).



Ces différences d'interprétation rendent l'intégration de design particulièrement ardue. Cette tâche, pouvant sembler mineure

aux yeux du profane, se révèle extrêmement coûteuse en ressources. Afin d'assurer une certaine productivité et de répondre aux exigences du marché, il est donc essentiel de faire appel à des spécialistes en la matière, experts de l'intégration de webdesign. L'ensemble des infographistes se sépare en deux voies quant au rendu de leurs travaux. Une première partie délivre au client un fichier contenant une image du site. Le développeur devra alors découper lui-même l'image, et la mettre en ligne. Cela implique de très solides connaissances de l'interprétation CSS des différents navigateurs. Cette solution, bien qu'avantageuse financièrement, est bien souvent plus désastreuse au niveau horaire/homme. Il est donc préférable de choisir le second type de rendu : le design, accompagné de l'intégration XHTML.

Cette dernière solution pose cependant un problème. En effet, un développement en Y est généralement décidé pour la réalisation d'un contrat, afin de faire patienter le moins possible la clientèle. Le design doit donc être intégré simultanément au développement. La solution à ce développement collaboratif est l'utilisation du modèle MVC (Model-View-Controller).

Le modèle MVC permet la séparation des différentes couches d'une application. On a tout d'abord la couche Model, qui gèrera les accès aux bases de données, calculera les différentes valeurs des variables, etc. C'est le développeur qui aura l'exclusivité de cette couche métier. La deuxième (couche View) contient tout ce qui sera vu par l'utilisateur final : formulaires, textes, images. C'est cette couche qui sera manipulée par le graphiste. Enfin, la dernière permet le contrôle des saisies utilisateurs (une adresse e-mail a-t-elle le bon format `xxxxxxx@xxxxxx.xxx` ?).

Le principe d'un moteur de templates est très simple : il relie la couche modèle à la couche vue. C'est ainsi qu'il gère le passage de variables entre ces deux couches. Par exemple, si on veut afficher un âge constamment mis à jour, d'une année sur l'autre. Le développeur calculera celui-ci selon la date récupérée d'une base de données. Ensuite, il transmettra au graphiste une variable que l'on pourra nommer `age_membre`. Celui-ci pourra alors, même s'il n'est pas familier avec le langage utilisé, afficher facilement l'âge du membre sans s'interroger sur la partie de code PHP correspondante. Par conséquent, l'intégration graphique d'un site est rendue bien plus aisée. D'autant plus si des conventions de nommage sont appliquées. Le développeur réalise toute la partie métier de l'application pendant que le designer applique celle-ci de son côté dans la partie graphique. Ce modèle, reconnu dans de nombreux autres langages

tels que Java à l'origine, a déjà fait ses preuves, et permet une intégration des plus rapides. De plus, le client peut, en direct, voir son site évoluer, et faire ses remarques au designer, qui pourra modifier en toute tranquillité son code sans pour autant interférer avec les codeurs de la couche métier. Mais outre l'avantage time-to-market de ce système, on accède à un autre avantage non négligeable : la séparation du code en ses différentes composantes, ce qui est de plus en plus dans l'ère du temps (et qui est aisé à comprendre en terme de pérennité du code). Il est en revanche important de parler des inconvénients d'un tel système. L'ajout d'un script supplémentaire liant les différentes parties ajoute forcément un délai lors de l'affichage de la page. Il est cependant possible de le réduire un maximum en optimisant son code, et en supprimant les fonctions inutiles. Le but de cet article sera donc de décrire la réalisation d'un squelette de moteur de templates, vous permettant d'appréhender différentes difficultés relatives à la création d'un tel objet.

## 1 LECTURE D'UN TEMPLATE

Commençons tout d'abord par créer notre classe TemplateEngine.

```
<?php
class TemplateEngine
{
}
?>
```

A présent, nous allons lire le contenu d'un fichier \*.html grâce à notre moteur. Pour cela, nous allons créer un attribut privé `$buffer` qui contiendra le contenu parsé de nos templates. Pour le remplir, nous ferons appel à une méthode `ParseFile` et pour l'afficher, à une méthode `Output`.

```
private $buffer;
public function __construct()
{
    $this->buffer = '';
}
public function ParseFile($filename)
{
    $filename = addslashes($filename);
    if(file_exists($filename))
    {
        $this->buffer = file_get_contents($filename);
    }
    else throw new Exception('Le fichier '.$filename.' n\'existe pas.');
```

```
public function Output()
```

```
{
    echo $this->buffer;
}
```

Ce code est très simple à comprendre. La méthode `ParseFile` prend en paramètre le nom du fichier à parser. On échappe automatiquement les caractères pouvant poser problèmes (notamment les `/`) afin de faciliter la tâche des développeurs. On se contente ensuite de mettre dans notre buffer le contenu du fichier. L'utilisation d'une exception est, bien entendu, facultative. Il est tout à fait possible d'afficher à la place un message d'erreur. Celle-ci n'est présente que si vous désirez faire de la remontée d'exceptions, dans le cas par exemple où vous auriez un gestionnaire d'erreurs spécifique. L'utilité de la méthode `Output` réside dans le fait qu'il est possible de changer la méthode de sortie très facilement. Par exemple, on peut appliquer une `utf8_encode` à tout un site en appliquant cette fonction une seule et unique fois sur le buffer. Testons à présent notre moteur. Pour cela, faisons un simple fichier `*.html` :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Ca marche !</title>
</head>
<body>
    <p>Notre moteur de templates fonctionne ! :)</p>
</body>
</html>
```

Appelons donc notre moteur de templates et sa méthode `Output` dans un script PHP.

```
<?php
require('templateEngine.php');
$tpl = new TemplateEngine();
$tpl->ParseFile('index.html');
$tpl->Output();
?>
```

Il ne nous reste alors plus qu'à tester notre page.

## 2 PASSAGE DE VARIABLES

Pour l'instant, l'utilité est plutôt négligeable. En effet, nous ne faisons qu'afficher une page statique. Dans ce cas, pourquoi ne pas afficher directement la page `*.html` ? Nous allons donc voir la première utilité de ce moteur : le passage de variables du PHP au code HTML, sans mélanger les deux. Pour cela, nous allons coder la méthode privée `SetVariables`. On décidera d'inclure nos variables dans notre template sous la forme `#{nom_variable}`.

```
private function SetVariables()
{
    # Récupération de toutes les variables du template.
    preg_match_all('#\${(.*?)}\#U', $this->buffer, $varNames, PREG_PATTERN_ORDER);
    $varNames = $varNames[1];
```

```
# Pour chaque variable, on remplace celle-ci par sa valeur.
foreach($varNames as $varName)
{
    global $$varName;
    $this->buffer = preg_replace('#\${(.*?)}\#U', $$varName, $this->buffer);
}
}
```

Ce code peut sembler complexe au premier abord, mais il n'en est rien. La seule difficulté réside dans le choix des expressions régulières. On place dans le tableau `$varNames` tous les noms de variables récupérés dans le template, en respectant la norme fixée (à savoir `#{nom_variable}`). On prend le second élément du tableau (indice 1), le premier étant un récapitulatif des occurrences vérifiant la regex passée en paramètre. Il ne faut pas oublier de rendre l'algorithme de recherche non gourmand en ajoutant l'option `U` (pour `ungreedy`) à notre expression régulière. En effet, si on le supprime, le parseur prendra en compte tout le texte compris entre la première et la dernière accolade, et non pas le texte entre chaque paire d'accolades.

## 3 SOUS-TEMPLATES

Nous allons maintenant découper notre template en plusieurs sous-parties. Cela peut-être utile pour, par exemple, afficher une liste de membres. Nous utiliserons une notation qui permet de bien séparer visuellement les différentes parties du template. Transformons donc notre template principal pour afficher une liste de membres.

```
<!--===== begin_template members =====>
[... ]
<ul>
    ${members }
</ul>
[... ]
<!--===== end_template =====>
<!--===== begin_template member =====>
<li>
    <strong>${firstName} ${lastName}</strong><br />
    ${title}
</li>
<!--===== end_template =====>
```

Pour gérer les sous-templates, il nous sera nécessaire de modifier notre moteur. Il faut qu'il soit capable de détecter les différents sous-templates, et de ne remplacer les variables que dans celui concerné, plutôt que de parcourir le fichier dans sa totalité. On ajoute tout d'abord un attribut `$filename` que nous initialisons dans le constructeur. Nous pouvons aussi supprimer l'attribut `$buffer`, qui ne nous sera plus d'aucune utilité, comme nous le constaterons plus tard.

```
private $filename ;
public function __construct($filename)
{
    $this->filename = addslashes($filename);
}
```

Modifions donc notre méthode `ParseFile` afin de prendre en compte les différentes particularités de nos sous-templates :

```

private $originalSubTemplates;
private $modifiedSubTemplates;
private function ParseFile()
{
    if(file_exists($this->filename))
    {
        $content = file_get_contents($this->filename);
        preg_match_all("#<!--===== begin_template (\w*) =====>
        >(.*)<!--===== end_template =====>#sU",
            $content, $matchedExp, PREG_PATTERN_ORDER);
        $subTemplatesNames = $matchedExp[1];
        $subTemplatesContent = $matchedExp[2];
        foreach($subTemplatesNames as $index => $subTemplateName)
        {
            $this->originalSubTemplates[$subTemplateName] = $subTem
            platesContent[$index];
        }
    }
    else throw new Exception('Le fichier '. $this->filename.' n\'
    existe pas.');
```

Afin d'optimiser un maximum notre moteur, nous garderons en mémoire dans les deux variables \$originalSubTemplates et \$modifiedSubTemplates respectivement les parties du template telles que parsées lors de la première lecture du template, et dans la seconde, le contenu du sous-template avec remplacement de toutes les variables. Pour l'expression régulière, il est essentiel de passer l'option `s` afin que le caractère universel (le point `.`) prenne en compte aussi les sauts de ligne. Il nous suffit maintenant de faire appel à cette méthode (devenue privée) dans notre constructeur, afin de mémoriser un tableau des différentes sous-parties de notre template. Modifions à présent notre méthode de liaison du code à la partie graphique :

```

private function SetVariables($TemplateName)
{
    # Récupération de toutes les variables du template.
    preg_match_all('#\$\\{(.*)\\}#U', $this->originalSubTemplates
    [$TemplateName], $varNames, PREG_PATTERN_ORDER);
    $varNames = $varNames[1];
    # Réinitialisation du template modifié, au cas où des traitements
    aient déjà eu lieu sur celui-ci.
    $this->modifiedSubTemplates[$TemplateName] = $this->original
    SubTemplates[$TemplateName];
    # Pour chaque variable, on remplace celle-ci par sa valeur.
    foreach($varNames as $varName)
    {
        global $$varName;
        $this->modifiedSubTemplates[$TemplateName] = preg_replace
        ('#\$\\{'. $varName. '\\}#', $$varName,
            $this->modifiedSubTemplates[$TemplateName]);
    }
}
```

La seule différence avec la méthode précédente est que l'on modifie les différentes variables dans l'attribut \$modifiedSubTemplates au lieu de la variable \$buffer. Il nous faut enfin une dernière méthode, qui nous permettra d'attribuer à une variable le contenu d'un sous-templa-

te. De plus, cette fonction nous servira de méthode d'accès à notre moteur (tous les arguments et méthodes étant désormais privés).

```

public function GetHTMLCode($TemplateName)
{
    $this->SetVariables($TemplateName);
    return $this->modifiedSubTemplates[$TemplateName];
}
```

Cette méthode, très simple, se contente d'attribuer à notre sous-template les variables correspondantes, et nous retourne le code HTML final. On optimise enfin un peu notre code en rendant la méthode `Output` statique : en effet, une méthode statique s'exécute environ quatre fois plus rapidement qu'une méthode classique. De plus, cela nous facilitera la programmation de notre classe, comme nous pourrions le constater dans l'exemple suivant.

```

public static function Output($buffer)
{
    echo $buffer;
}
```

## 4 EXEMPLE

Voyons à présent un exemple pratique de notre classe permettant l'affichage d'une liste de membres.

```

<?php
require('templateEngine.php');
$tpl = new TemplateEngine('index.html');
# Récupération des données, habituellement provenant d'une
base de données SQL.
$members_table = array(    array( 'Jonathan', 'Petitcolas',
'Web Lab. member'),
        array( 'Mauricio', 'Diaz Orlich', 'Web Lab. director'),
        array( 'Benjamin', 'Bouché', 'Web Lab. SCT'),
    );
foreach($members_table as $member)
{
    $firstName = $member[0];
    $lastName = $member[1];
    $title = $member[2];
    $members .= $tpl->GetHTMLCode('member');
}
TemplateEngine::Output($tpl->GetHTMLCode('members'));
```

Le code est suffisamment explicite de lui-même pour se passer de commentaires. On remarquera cependant que l'on concatène les différents éléments de la liste de membres, conformément au modèle défini dans notre template. A présent, nous disposons d'un moteur de templates des plus basiques. De nombreuses autres fonctionnalités restent à implémenter, selon vos propres besoins. Par exemple, il pourrait être intéressant de mettre une gestion des templates sur plusieurs fichiers, ou encore d'implémenter l'insertion de données émanant de tableaux avec une syntaxe simplifiée du type `$(tab[0])`.



■ Jonathan Petitcolas

Laboratoire des technologies WEB - SUPINFO Montréal  
Gérant d'Imagiweb - <http://www.imagiweb.fr>



# Développer à l'aide de composants

Malgré la multitude d'applications web présentes sur la toile ou sur les intranets des entreprises, l'analyse de celles-ci démontre que leur contenu est parfois semblable (Flux RSS, Calendrier, Formulaire de contact...). Faut-il pour autant réécrire le code à chaque nouvelle application ? A travers cet article nous verrons que les composants pourront nous éviter cette fastidieuse réécriture.

**i**maginons une application web proposant l'affichage d'un calendrier. A première vue, rien de très compliqué, mais estimons tout de même le temps de développement à une demi journée. Quelques semaines plus tard, nous devons développer une autre application proposant également un calendrier. Le développement devra alors être réitéré. Certes, le temps réalisation sera légèrement réduit, mais cela aura un coût qui aurait pu être évité si l'on avait utilisé un composant.

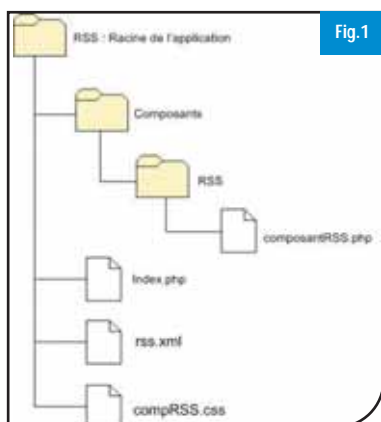


Fig.1

## MISE EN PLACE D'UN COMPOSANT

Nous allons développer en PHP un composant capable d'afficher un flux RSS. Ce composant recevra en paramètres le lien du flux RSS à afficher et le style CSS permettant la mise en forme de l'affichage.

### Arborescence de l'application

L'arborescence de l'application est détaillée en Fig.1.

Vous trouverez sur le cd Rom, ou notre site web, [www.programmez.com](http://www.programmez.com) le fichier rss.xml contenant le flux rss à afficher (issu du site définissant le standard RSS 2.0)

### Création du composant

Nous allons créer une classe et le constructeur de celle-ci (fichier composantRSS.php) :

```

class agregateurRSS
{
    var $url; //Url du flux RSS
    var $styleCSS; //Url du style CSS a utiliser

    //Constructeur
    function agregateurRSS($paramUrl,$paramStyle)
    {
        $this->url = $paramUrl;
        $this->styleCSS = $paramStyle;
    }
}
  
```

La méthode principale va consister à analyser les items du flux RSS et à créer un tableau qui sera envoyé à une méthode annexe chargée de mettre en forme les données analysées :

```

function lecture($paramUrl,$paramStyle)
{
    $ligne = file_get_contents($paramUrl);
    //Tableau de données
    $fluxRss[]=array();

    //Décomposition du Flux : Analyse des items
    if( eregi("<item>(.*?)</item>", $ligne, $tableauItems) )
    {
        $items = explode("<item>", $tableauItems[0]);
        $nbItems = count($items)-1;

        for ($i=0;$i<$nbItems;$i++)
        {
            eregi("<title>(.*?)</title>", $items[$i+1], $titre);
            eregi("<category>(.*?)</category>", $items[$i+1], $
            categorie);
            eregi("<description>(.*?)</description>", $items
            [$i+1], $description);
            eregi("<link>(.*?)</link>", $items[$i+1], $lien);
            eregi("<author>(.*?)</author>", $items[$i+1],
            $auteur);
            eregi("<pubDate>(.*?)</pubDate>", $items[$i+1],
            $date);
            if( eregi("<![CDATA\[([.*])\]>", $description
            [1], $cdata) )
            {
                $description[1]=$cdata[1];
            }
            else
            {
                $description[1]=$description[1];
            }
            if( eregi("<p>(.*?)</p>", $description[1], $cdata) )
            {
                $description[1]=$cdata[1];
            }
            else
            {
                $description[1]=$description[1];
            }
        }
    }
}
  
```

```

    }

    //Ajout des données au tableau principal
    $s = sizeof($fluxRss);
    $fluxRss[$s]= array($titre[1],$categorie[1],
    $description[1],$lien[1],$auteur[1],$date[1]);

    }

    //Affichage du flux (Appel de la méthode de mise en forme)
    $this->afficheFlux($fluxRss,$paramStyle);

    }
}

```

La dernière étape consiste à créer une méthode de mise en forme des données :

```

public function afficheFlux($fluxRss,$styleCSS)
{
    for ($i=1;$i<sizeof($fluxRss);$i++)
    {
        //Appel du style CSS
        echo "<link rel='stylesheet' href='". $styleCSS. "'
type='text/css' />";

        //Affichage des résultats
        echo "<div id='canal_rss'>";
        echo "<div id='titre_rss'>". $fluxRss[$i][0]. "</div>";
        echo "<div id='description_rss'>".html_entity_decode($flux
Rss[$i][2]). "</div>";
        echo "</div>";

    }
}

```

## MISE EN PLACE DU COMPOSANT

Tout d'abord, éditons le fichier compRSS.css afin de définir le style d'affichage.

```

#canal_rss{
    width:300px;
}

#titre_rss{
    font-family:Verdana;
    font-size:11px;
    background-color:#000000;
    color:#ffffff;
}

#description_rss{
    font-family:Verdana;

```

```

font-size:11px;
background-color:#FFFFFF;
color:#000000;
margin-bottom:10px;
}

```

Enfin, utilisons notre composant dans la page index.php:

```

!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset
=ISO-8859-1" />
</head>
<body>
<?php

//Appel du composant
require 'composants/RSS/composantRSS.php';

//Lecture du fichier et affichage du résultat
$agregateur = new agregateurRSS("rss.xml","compRSS.css");
$agregateur->lecture($agregateur->url,$agregateur->styleCSS);

?>
</body>
</html>

```



Mise en place du composant

## CONCLUSION

On constate que la mise en place du composant est simplissime. C'est à cette étape de ce type de développement qu'il tire tous ses avantages. La réutilisabilité de notre composant est tout à fait possible dans une autre application, grâce à son paramétrage, évitant ainsi la réécriture du code. Néanmoins, il convient au développeur de porter toute son attention sur l'écriture de son composant en veillant à prendre en considération la gestion des exceptions de la fonction développée (Ex : Gestion du nombre de jours du mois de février dans un composant calendrier...). Un composant doit être autonome et ne doit nécessiter aucune modification (ou minimales) de son utilisateur, ce qui n'est pas du tout le cas pour notre composant ! Je vous laisse donc le soin de l'améliorer.



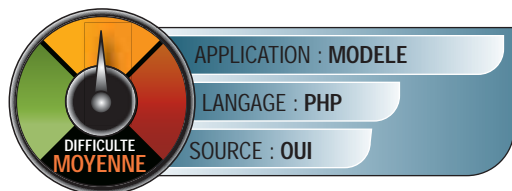
■ Aurélien Vannieuwenhuyze  
Ingénieur Concepteur / Réalisateur  
Sylis France ([www.sylis.com](http://www.sylis.com))  
Blog : <http://aurelienv.no-ip.org>

# UML et le génie logiciel : développer avec PHiMX

Sur le CD  
du numéro !



**Unified Modeling Language** permet de modéliser une application au sein d'un système d'information. Ces modèles guident l'équipe d'architecture et de développement lors des phases d'analyse, de conception et de tests. Afin que tout le monde autour d'un projet puisse s'entendre, on comprend bien qu'un modèle normalisé s'impose.



A partir de ce modèle, que l'on pourra manipuler dans tous les sens, il est alors possible de générer une grande partie du code

du projet, mais il est aussi possible de faire l'inverse : des fichiers de code source peuvent être analysés pour régénérer les diagrammes et schémas UML.

Le code en question peut être écrit dans n'importe quel langage, en revanche il est presque obligatoire qu'il soit orienté objet. L'orientation objet du code permet un découpage efficace de l'application en entités propres et uniques qui ont des rôles très précis. C'est aussi un moyen tout à fait adapté au travail à plusieurs sur un projet, chaque personne pouvant s'occuper d'une fonction du logiciel.

UML est la phase préliminaire à la programmation proprement dite. C'est elle qui permet de modéliser l'étude théorique d'une application. En génie logiciel, on pratique la découpe fonctionnelle d'un système : cela consiste à factoriser certains comportements communs du logiciel. En d'autres termes : pour réaliser une fonction précise du logiciel, on peut utiliser un ensemble d'autres fonctions, déjà disponibles, si l'on a pris la peine de rendre ces dernières un tant soit peu génériques. Factoriser les comportements peut en revanche avoir de lourdes conséquences. Les fonctions devenues interdépendantes, une simple mise à jour à un endroit donné, peut impacter en cascade une multitude d'autres fonctions.

On écrit donc moins de code (Don't Repeat Yourself : n'écrivez pas 2 fois le même paradigme), ainsi statistiquement, on a moins de risque d'introduire un bug. En contrepartie, le code écrit possède beaucoup de responsabilités, car il est utilisé dans de nombreuses fonctions du logiciel final; ainsi, s'il venait à se comporter de manière hasardeuse, l'application entière en serait fortement impactée.

UML aide donc à la conception de composants logiciels robustes, extensibles et réutilisables. Pour créer de tels composants, il faut s'efforcer de lutter contre certains risques pénalisant les activités de développement :

**La rigidité** : chaque évolution est susceptible d'impacter de nombreuses parties de l'application. Le développement est de plus en plus coûteux, ce qui introduit des risques au sein même du développement. Le coût des modifications étant élevé, le logiciel a peu de chances d'évoluer après sa mise en production.

**La fragilité** : la modification d'une partie de l'application peut provoquer des erreurs dans une autre partie de l'application. Le logiciel est peu robuste, et le coût de maintenance reste élevé. Les modifications étant de plus en plus risquées, les chances d'évolution du

logiciel après son passage en production sont là aussi, amaigrées. La conduite du changement est une discipline à part entière sur laquelle de nombreux livres ont été écrits.

**L'immobilité** : il est difficile d'extraire une partie de l'application pour la réutiliser dans une autre. Le coût de développement de chaque application reste élevé puisqu'il faut repartir de zéro à chaque fois. La programmation orientée objet, lorsque bien utilisée, permet justement d'éviter ce genre de situation. L'utilisation de frameworks aussi, répond tout à fait à cette problématique.

Enfin, **la viscosité** : lors des changements, les développeurs ont souvent plusieurs façons de procéder. Certaines méthodes préservent la conception de départ, d'autres non. Celles préservant la conception sont souvent plus difficiles à mettre en oeuvre. Les développeurs préfèrent donc ne remplacer qu'une petite partie sans préserver la cohérence du code de départ. Le désordre s'installe alors, le logiciel se complexifie au fur et à mesure, jusqu'à finalement devenir inmaintenable. Ces quatre symptômes sont les signes flagrants d'une mauvaise architecture logicielle. Une application qui présente ces symptômes risque fort de dégénérer rapidement.

## LE CAS PHP

PHP5 est une plate-forme qui possède un modèle objet complet permettant d'utiliser pleinement UML et les principes de conception logicielle. Avec la fin du support de PHP4, l'année 2008 va être marquée par une forte migration vers PHP5, et c'est tant mieux pour la pérennité des applications conçues ! PHP5 permet réellement de créer des architectures pleinement orientées objet, qui n'ont rien à envier à leurs homologues écrites en Java. La multiplication des frameworks PHP5 très professionnels, tels que ZendFramework ou Symfony, montrent que cette tendance est à la hausse. Dès lors, lorsqu'on parle de PHP, il s'agit de la version 5.

## LOGICIELS POUR FAIRE DE L'UML

Pour PHP malheureusement, peu de logiciels permettent actuellement de générer du code, bien qu'ils puissent tous modéliser une application PHP. Nous pouvons citer l'un d'entre eux : DIA. Disponible sous toutes plates-formes, DIA est basé sur GTK : il est libre, open-source, et très léger. Il permet de modéliser de l'UML et de générer du code compatible PHP5 (à partir d'un plug-in). Toujours dans l'open-source : StarUML. Celui-ci, tournant uniquement sous Windows, ressemble à Umbrello sous Linux.

## RÉTRO-INGÉNÉRIE

Tous les logiciels UML utilisent leur propre format de données. Cependant, ils lisent tous (ou presque), un format dérivé du XML : le



Sous le haut patronage du Ministère de l'Economie, de l'Industrie et de l'Emploi



Fédération Nationale de  
l'Industrie du Logiciel Libre

PRÉSENTE

24 et 25 septembre 2008

# Paris Capitale du Libre



Informations et inscription gratuite sur

**[www.paris-libre.org](http://www.paris-libre.org)**

01

CIO

LeJournalduNet

**LiNux+**

PC  
EXPERT

phpsolutions

**Pro**grammez!

01net.

La Tribune

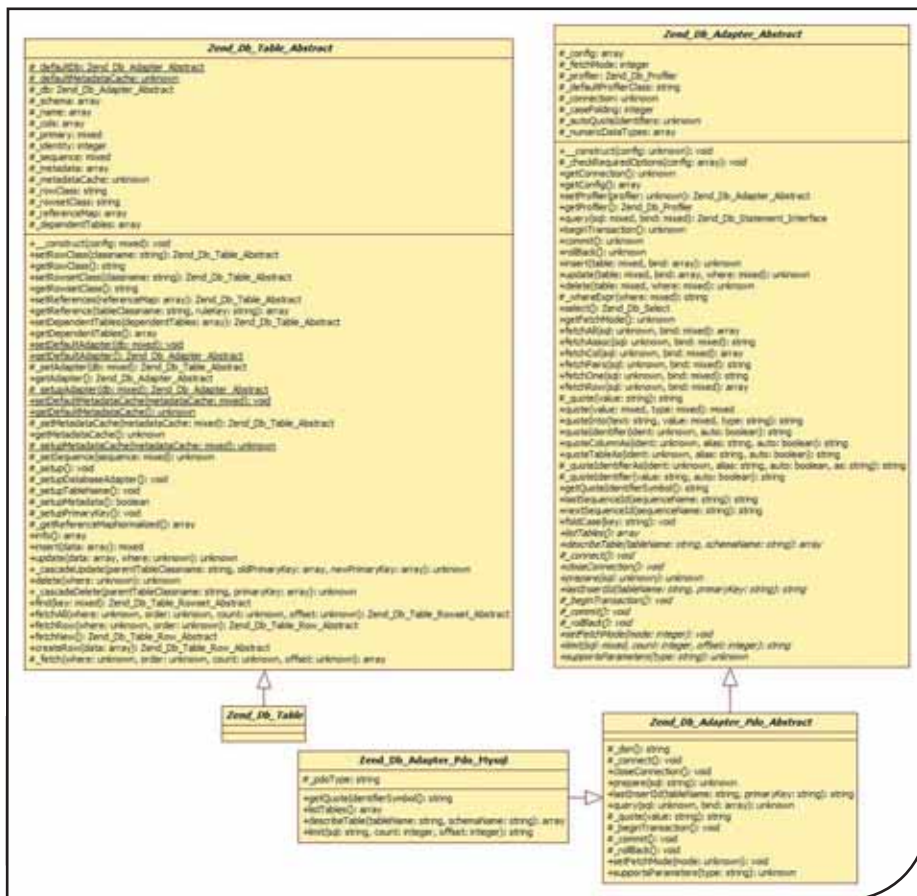
Le Monde Informatique

Mi  
musée de  
l'informatique

Progilibre.com

toolinux▶

L'événement européen  
dédié à GNU/Linux et aux logiciels libres  
à la **Maison Internationale de Paris**



XMI (XML Metadata Interchange). Ainsi, si un programme est capable d'analyser un code source, et de générer un fichier XMI, alors on a effectué de la rétro-ingénierie : on part d'un code source pour générer un fichier XMI, qui va décrire des classes, dans des logiciels de modélisation UML. Or, PHP5 propose tous les outils pour ceci : un modèle objet complet, et une API de *réflexion*. Celle-ci permet d'introspecter le modèle de PHP5 : regarder comment une classe est fabriquée, et en extraire des données. D'un autre côté, les possibilités de PHP5 envers le XML sont très avancées...

## PHiMX

PHiMX est un composant (un ensemble de classes écrites en PHP), qui permet de générer de l'UML à partir d'un code PHP. Il est open-source et utilisable gratuitement.

PHiMX est un package qui existe dans PEAR. Il est aussi téléchargeable hors PEAR. Vous pouvez l'installer via la commande :

```
pear install http://phimx.sourceforge.net/PHiMX-1.0b1.tgz
```

Si vous ne possédez pas PEAR, vous pouvez télécharger le package hors PEAR ici :

[http://sourceforge.net/project/showfiles.php?group\\_id=213025](http://sourceforge.net/project/showfiles.php?group_id=213025)

## UTILISATION EN LIGNE DE COMMANDES

Nous partons d'une installation PEAR. C'est plus pratique car elle est intégrée au PATH du système : en ouvrant une ligne de commande et en tapant simplement `phimx --help`, nous voyons apparaître les options (quelques warnings PHP sont à noter, PEAR étant

encore en grande partie sous PHP4, mais ceci ne gêne en rien l'utilisation de `phimx`)

`phimx -d <directory>` va scanner tout un répertoire pour générer un fichier XMI relatif aux classes présentes dans ce répertoire.

`phimx -r -d <directory>` scanne récursivement les répertoires.

Il y a encore quelques autres options que je vous laisse découvrir par la pratique :-)

## NOTRE EXEMPLE : LE PACKAGE ZEND\_DB

Zend\_Db est un ensemble de classes (un package) faisant partie du Zend Framework. Il s'agit du package qui s'occupe de la liaison avec la base de données. Nous allons utiliser très simplement PHiMX afin de générer des diagrammes UML du package, en partant donc des sources PHP :

```
F:\> phimx -r -d D:\www\ZF\trunk\library\Zend\Db Zend_Db.xmi
```

PHiMX commence alors à analyser le dossier `D:\www\ZF\trunk\library\Zend\Db`, récursivement, comme je l'ai demandé, et il génère ensuite une sortie dans `F:\Zend_Db.xmi` ( je suis parti de `F:\`) Notez qu'il faut que Zend Framework soit ajouté à votre `include_path` à cause des dépendances, pour

moi c'est déjà le cas dans mon `php.ini`

## ANALYSE DU XMI

Maintenant, en ouvrant le XMI avec startUML (importation), j'obtiens toutes mes classes, qu'il me reste simplement à tirer sur ma feuille de diagramme de classes : J'ai toutes mes classes à disposition, via le XMI, à moi d'utiliser celles que je veux. J'en ai modélisé peu ici, car elles sont assez grosses comme vous pouvez le remarquer...

On notera que tous les liens d'héritage ou d'implémentation d'interface sont conservés. De même, les attributs et méthodes sont bien représentés, avec, la plupart du temps, les bons types. En revanche, les utilisations (aggrégation, composition) ne sont pas conservées.

## CONCLUSION

Les possibilités objet de PHP5 font qu'il devient possible, voire indispensable, d'utiliser des outils tels que UML dans ses projets. PHiMX est un programme de rétro-ingénierie très intéressant. L'approche par rétro-ingénierie est très pratique pour modéliser un framework, ou un composant que l'on utilise, sans l'avoir écrit, ce qui est souvent le cas en PHP5.

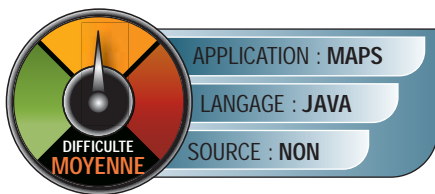
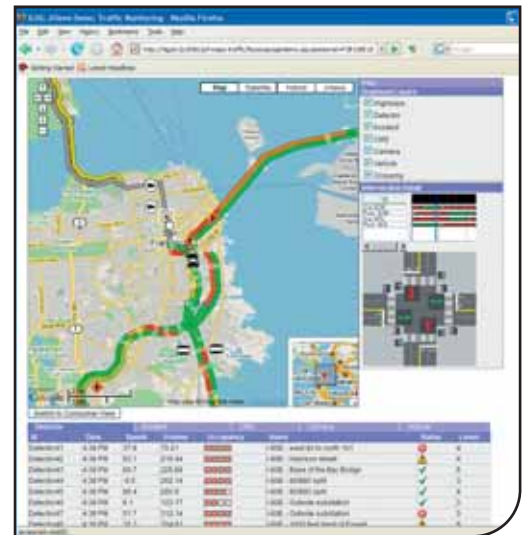
■ Julien PAULI

chargé des formations PHP et ZendFramework chez Anaska (groupe AlterWay). Il pratique le génie logiciel et utilise UML dans le cadre de ses développements. Il est membre fondateur de la communauté francophone du ZendFramework (<http://www.z-f.fr>).



# Combiner Google Maps et composants JSF

Beaucoup considèrent Google Maps comme un des meilleurs outils de cartographie en ligne car il est intuitif et bien conçu. Il fournit des photographies aériennes avec un détail impressionnant pour de nombreuses régions et des informations au niveau de la rue pour toutes les grandes régions métropolitaines. Nous allons voir comment [utiliser Maps et des composants JSF](#).



Une des questions que l'on peut se poser est : " comment puis-je utiliser cet excellent outil avec mon serveur Java EE qui sert un contenu dynamique ? "

## LES SUPERPOSITIONS (" OVERLAYS ") GOOGLE MAPS

Maps possède une bibliothèque AJAX performante. Le plus intéressant est que cette bibliothèque fournit plusieurs techniques de superpositions personnalisées des contenus par dessus les cartes habituelles des rues ou les images satellites.

- **Marqueurs, lignes et zones** : Vous pouvez créer - à la volée - des points sur la carte en utilisant des marqueurs (parfois avec une icône personnalisée). Vous pouvez également ajouter des lignes et des zones qui seront affichées sur la carte avec la couleur que vous définissez.
- **Superposition de tuiles** : L'API contient un mécanisme pour utiliser une grille d'images qui résident (ou sont générées dynamiquement) sur un serveur, sur la base d'une simple syntaxe URL. Ce mécanisme peut être utilisé pour produire des requêtes spécifiques qu'un serveur Web adapté peut comprendre.
- **Superpositions Personnalisées** : Elles peuvent être utilisées si vous avez besoin d'encore plus de contrôle, au niveau du DOM, sur les éléments HTML qui doivent être créés.
- **Superpositions "Sol" (Ground Overlays)** : Ce sont des zones rectangulaires qui peuvent être utilisées pour géo-référencer une seule image, à un endroit précis.
- **Superpositions KML** : Vous pouvez passer l'URL accessible au public des fichiers KML pour afficher des images géo-référencées, des polygones, des lignes ou des marqueurs.

## COMMENT UTILISER CELA ?

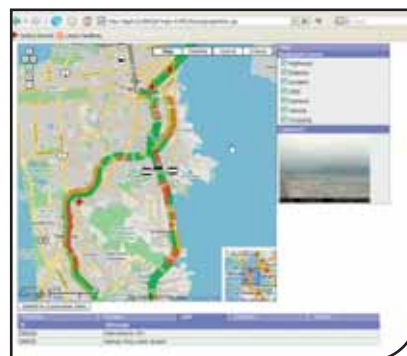
En raison de la restriction à l'utilisation des superpositions KML (qui doivent être accessibles sur un serveur public), ce type de couverture est difficile à utiliser lorsque votre serveur doit générer dynamiquement un fichier. Il est de plus difficile à développer, car, dans la plupart des cas, les développeurs ont leur propre serveur Web local que Google Maps ne peut pas voir, et refuse d'utiliser comme source KML. J'utiliserais les superpositions personnalisées uniquement

dans les cas les plus avancés, quand tout le reste échoue, car elles nécessitent une connaissance approfondie de la façon dont le HTML dynamique fonctionne à l'intérieur de Google Maps.

Les superpositions " Sol " sont plutôt faciles à utiliser, si vous êtes conscient du fait que Google Maps est un système de coordonnées basé sur une projection Mercator. Si votre image est dans une autre projection, vous devrez traiter et probablement re-projeter votre image, ce qui n'est pas une simple affaire. Si vous acceptez un certain niveau de déformation, ou votre image est "petite" par rapport à la surface terrestre, vous pouvez utiliser des images qui sont en coordonnées latitude / longitude. Dans ce cas, l'effet principal de la projection Mercator sera d'étirer l'image verticalement.

Marqueurs et polygones peuvent être créés à la volée, par l'intermédiaire d'une API simple, en fonction d'informations récupérées depuis le serveur. C'est assez simple, mais il faudra un peu de développement java script. Si vous avez beaucoup de données vous aurez probablement à utiliser des techniques Ajax pour détecter ceux qui sont visibles sur la carte, et les créer de manière asynchrone. Si votre serveur est en mesure de générer des grilles d'images dans une projection Mercator, la superposition de tuiles est probablement la méthode la plus polyvalente. Elles vous permettent de " quadriller " la carte de manière asynchrone lorsque l'utilisateur zoome ou se déplace sur la carte. Si vos données changent rapidement, comme Google Maps met en cache les images reçues, vous risquez peut-être d'afficher des tuiles incompatibles entre elles à un moment donné. Ce type de couverture est donc la meilleure solution pour l'affichage d'informations qui ne changent pas (ou très lentement). Comme en général c'est une action utilisateur qui déclenche la demande de la tuile, cela rend plus difficile le contrôle des flux de

données par le serveur. Les tuiles étant typiquement au-dessus des couches par défaut de Google Maps, vous aurez besoin de générer des PNG (ou GIF) avec des parties transparentes (ou semi-transparentes), si vous voulez voir celles-ci à travers vos tuiles.





## CODE JAVASCRIPT D'UN EXEMPLE D'INTÉGRATION

Utiliser une superposition de tuiles pour afficher des images du serveur

Dans un premier temps, créons une classe Java Script, basée sur les prototypes, qui récupère côté serveur les tuiles pour les afficher sur notre navigateur. Cette classe nécessite un identificateur d'un conteneur HTML "DIV ", pour indiquer ou positionner la carte sur la page, comme l'exige l'API de Google Maps. Voici un exemple de code pour ajouter une superposition de tuiles personnalisées sur cette carte, et pour récupérer les images sur un serveur :

```
function GoogleMapView (parentDiv) {
  this.parentNode = document.getElementById(parentDiv);
  this.map = new GMap2(this.parentNode);
  // positionner les copyrights, au besoin
  var copyCollection = new GCopyrightCollection('My Company');
  copyCollection.addCopyright(new GCopyright(1,
    new GLatLngBounds(
      new GLatLng(-90, -180),
      new GLatLng(90, 180)), 0, "@2007"));
  // créer la couche superposée
  this.tiledLayer = new GTileLayer(copyCollection, 1, 17);
  // Créer la fonction qui construit l'adresse des tuiles.
  var client=this;
  this.tiledLayer.getTileUrl = function(p0,zl) {
    return client.getTileUrl(p0,zl);
  }
  // ajouter la superposition
  this.map.addOverlay(new GTileLayerOverlay(this.tiledLayer))
}
```

Nous aurons besoin d'une méthode qui fixe les URL de base pour atteindre le serveur (qui sera appelé lors du chargement de la page) telle que :

```
GoogleMapView.prototype.setServletURL = function (servletURL) {
  this.servletURL = servletURL;
};
```

Nous devons également avoir une méthode utilitaire qui calcule une chaîne de caractères à partir de deux points géo-référencés. Par exemple :

```
GoogleMapView.prototype.getBox= function (ne,sw) {
  return ne.getLat()+" "+ne.getLon()+" "+sw.getLat()+" "+sw.getLon();
};
```

Cela fait, nous pouvons écrire la méthode getTileUrl, utilisée ci-dessus, qui créera la requête Web

```
GoogleMapView.prototype.getTileUrl = function(p0,zl) {
  // calculer certains paramètres à partir du point central et
  // du niveau de zoom
  var proj=this.map.getCurrentMapType().getProjection();
  // coordonnées en haut à gauche
  var p1=new GPoint((p0.x+0)*256,(p0.y+0)*256);
  var latlon1=proj.fromPixelToLatLng(p1, zl);
  // coordonnées en bas à droite
```

```
var p2=new GPoint((p0.x+1)*256,(p0.y+1)*256);
var latlon2=proj.fromPixelToLatLng(p2, zl);
// Créer l'URL pour que le serveur retourne une image
// entre latlon1 et latlon2
var reqBuf = this.servletURL + '?request=image&bbox=' ;
reqBuf = reqBuf + this.getBox(latlon1,latlon2);
return uu;
}
```

Si vous avez un serveur qui est en mesure de retourner (en projection de Mercator) une image PNG quand il reçoit la requête <http://my.servlet/someUrl?request=image&bbox=N, S, E, W>, vous verrez votre propre couche de tuiles apparaître dans le composant Google Maps.

### L'ajout de marqueurs à la demande

Avec l'exemple ci-dessus, la carte Google passera par l'appel de méthode getTileUrl à chaque fois qu'une tuile n'est pas dans le cache et doit être affichée.

Pourquoi ne pas utiliser cela pour demander au serveur quels marqueurs doivent être ajoutés à la région ?

```
GoogleMapView.prototype.getTileUrl = function(p0,zl) {
  ...
  this.updateMarkers(latlon1,latlon2);
}
```

La méthode updateMarkers que nous avons introduite ci-dessus n'a plus qu'à envoyer une requête AJAX au serveur, qui devra lui retourner la liste des marqueurs appartenant à cette tuile.

```
GoogleMapView.prototype.updateMarkers =
function updateMarkers(layers,ne,sw) {
  // met à jour des marqueurs pour les symboles dans les bornes fournies
  var requrl=this.servletURL + '?request=markers&bbox='
  requrl=requrl + this.getBox(latlon1,latlon2);
  // Construire une demande AJAX
  var request = GXmlHttpRequest.create();
  request.open("GET", requrl, true);
  var client=this;
  request.onreadystatechange = function() {
    client.updateMarkers(request);
  }
  request.send(null);
}
```

Nous sommes maintenant prêts à décoder les informations venant du serveur, de manière asynchrone. Si l'on suppose que le serveur renvoie un tableau JSON contenant des informations sur les marqueurs.

```
GoogleMapView.prototype.updateMarkers = function (request) {
  if (request.readyState == 4) {
    var markers=eval("(" +request.responseText+")");
    for (i=0;i<markers.length;i++) {
      this.updateMarker(markers[i]);
    }
  }
}
```

Si nous supposons, pour simplifier, que chaque marqueur est uniquement identifié par un attribut "sid", nous pouvons mettre à jour notre liste de marqueurs.

```

GoogleMapView.prototype.updateSymbol=function (info) {
  // this.markers est le tableau des marqueurs connus
  var id=info.sid;
  if(this.markers[id]){
    // enlever l'ancien marqueur
    this.map.removeOverlay(this.markers[id]);
  }
  // créer le marqueur
  this.markers[id]=this.createMarker(info);
  if(this.symbols[id]){
    // add this new marker on the map
    this.map.addOverlay(this.markers[id]);
  }
}

```

Nous allons enfin créer le marqueur avec certaines API Google Maps, en veillant à ce que nous utilisions les informations du serveur. Par exemple les attributs "yLat" et "xLon" et "title".

```

GoogleMapView.prototype.createMarker = function (info){
  var sicon = new GIcon(G_DEFAULT_ICON);
  var loc=new GLatLng(info.yLat,info.xLon);
  var marker=new GMarker(loc, {
    icon:sicon,
    title:unescape(info.title),
  });
  return marker;
};

```

## ETAPES SUIVANTES

La méthode createMarker peut être améliorée pour changer l'image associée au marqueur (éventuellement au moyen d'un aller-retour au serveur, mais si vous avez beaucoup de marqueurs, vous devez prendre soin de ne pas surcharger celui-ci). Vous pouvez également installer des méthodes qui interceptent les événements souris pour quelque chose lorsque l'utilisateur clique sur le marqueur. Quelques méthodes Javascript supplémentaires vous permettraient également de gérer des couches séparées pour les marqueurs, afin d'afficher ou effacer indépendamment les superpositions, ce qui est souvent demandé dans les applications cartographiques.

## JAVA SERVER FACES

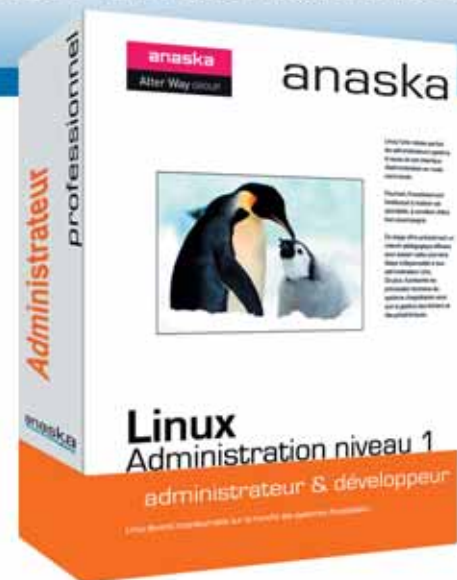
Il est assez simple d'envelopper tout cela dans un seul composant JSF qui expose seulement les balises nécessaires (comme le point central initial ou la base des URL de requêtes), et faire en sorte que le contexte côté serveur et côté client soient en phase, pour une utilisation facile dans n'importe quelle application.

### Quelques références

The Java BluePrints Map viewer component :  
<https://blueprints.dev.java.net/complib/v2/map-viewer.html>  
 The Google Maps API  
<http://www.google.com/apis/maps/documentation/>

os> Linux

## Formation Linux Administration



"Être à même d'administrer des serveurs sous Linux"

### Au Programme des 5 jours :

- \* Installer et configurer Linux
- \* Administrer Linux en ligne de commande
- \* mettre en place et utiliser des scripts shell
- \* Les bases d'une stratégie de sécurité
- \* Gérer les utilisateurs et les droits
- \* Gérer les applications
- \* Configurer le réseau

**10 % de réduction**  
pour  
les lecteurs de  
programmez

### Prochaines sessions

**Paris 2008**  
29 Septembre, 20 Octobre,  
17 Novembre

**Lyon 2008**  
22 Septembre  
17 Novembre

**Tarif**  
2250 € HT

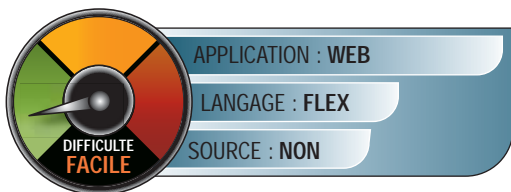
LE SPECIALISTE DE LA FORMATION POUR L'OPEN SOURCE



**Informations**  
01 45 28 09 82  
[www.anaska.com](http://www.anaska.com)

# Flex 3 compatible Linux ?

Jusqu'à présent, pour être capable de faire du développement Flex sous Linux, il fallait savoir se contenter d'un simple éditeur texte (vi par exemple ;) et utiliser le SDK afin de compiler son application et construire manuellement son archive à déployer. Adobe fournit maintenant de quoi créer et déployer ses applications tout en gardant son système favori.



Il est possible de télécharger sur le site d'Adobe trois outils :

- **Flex Builder 3** : l'outil de développement Flex basé sur Eclipse

- **Adobe Air** : le runtime permettant de faire tourner des applications RIA (Flex + HTML + PDF) sur un desktop
- **LiveCycle Data Services** : le framework de connexion de l'interface utilisateur à un backend Java.

Pour la partie Flex Builder, ce n'est pas une grande nouveauté puisque les alphas précédentes étaient déjà disponibles au téléchargement. En revanche, la disponibilité de AIR sous Linux est quant à elle entièrement nouvelle. Pour ce qui est de l'installation, les outils se présentent sous forme de binaires qu'il suffit d'exécuter.

## FLEX BUILDER 3

La version Linux de Flex Builder 3 offre sensiblement les mêmes possibilités que la version Windows à quelques différences près. D'abord, le design graphique dans l'éditeur MXML est absent. Ceci n'est pas forcément handicapant toutefois, car la plupart des développeurs utilisant Flex de façon intensive n'y ont pas recours. Plus ennuyeux, il n'est pas possible de faire du profilage CPU ou mémoire. Cette fonction n'est tout simplement pas implémentée et donc non disponible aussi bien dans les préférences qu'au niveau lancement du profilage. Outre ces limitations, des problèmes, pourtant déjà identifiés sous Windows, se retrouvent sous Linux

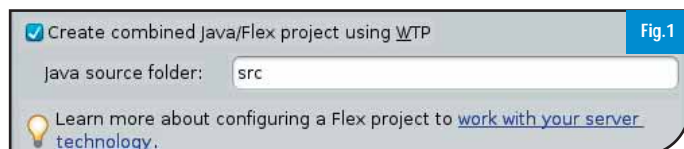


Fig.1

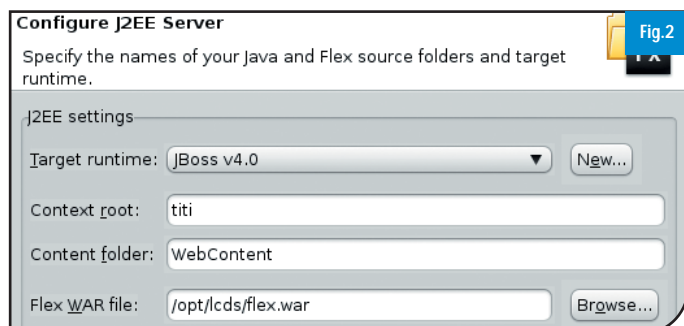
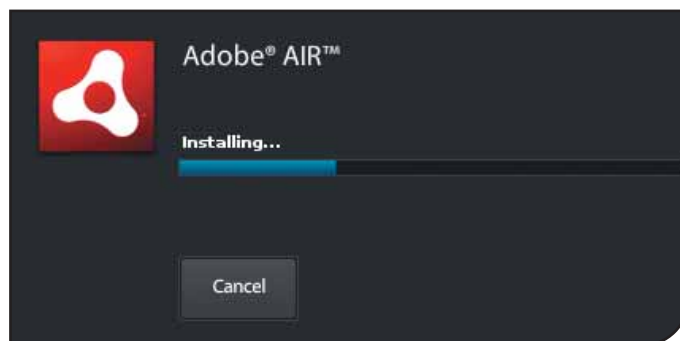


Fig.2



à savoir quelques manques notamment au niveau de la vue Flex Navigator qui ne permet pas de filtrer le contenu (par exemple pour ne voir que certains 'Working Sets' ou encore de ne voir que les projets ouverts). De même persiste un autre problème (nettement gênant) : la vue n'affiche pas les problèmes de compilation à l'aide d'une petite croix comme c'est le cas pourtant dans la vue Package Explorer. Mais tout n'est pas gris et on peut découvrir avec surprise la possibilité de créer des projets Web avec WTP intégrant la nature Flex. Cette nature vient se tisser sur le projet et peut être rajoutée après création d'un projet WTP standard. L'ajout de la nature s'appuie sur un WAR flex qui fonctionne déjà (Je me suis personnellement appuyé sur celui livré dans LiveCycle Data Services) afin de récupérer toutes les bibliothèques de dépendances et autres fichiers de configuration (attention cependant car les fichiers présents sont pour la plupart inutiles dans le cadre d'applications classiques). Le déploiement de l'application est de ce fait simplifié par l'utilisation du dépoyeur WTP. Par contre, toute modification d'un quelconque fichier ActionScript ou tout autre fichier relié à la compilation du swf final entraîne la re-compilation et le redéploiement du projet. Il faudra donc désactiver dans ce cas la construction automatique du projet sous Eclipse. Le debug fonctionne comme il se doit et il est possible de naviguer sans problème dans les sources (exception faite bien sûr des sources LiveCycle).

Malgré tout, l'outil est parfois instable (extinctions inopinées d'Eclipse) en plus d'être gourmand en mémoire. J'ai dû passer la taille maximale de la mémoire de la machine virtuelle d'Eclipse à 1 Go et passer le Perm Size à 512 Mo sans quoi, point de compilation. Finalement Flex Builder 3 est intéressant pour le réel gain de productivité qu'il apporte et le confort de travail. [Fig.1 et 2]. Sur Live Cycle Data Services, rien de nouveau, identique à la version Windows.

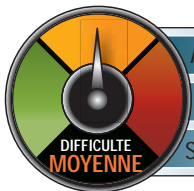
## ADOBE AIR

La disponibilité de Air est l'aspect de loin le plus intéressant. On sent un réel travail des équipes Adobe pour délivrer un logiciel de qualité et ne pas rater l'objectif, à savoir la possibilité de faire tourner une application desktop de façon fiable que ce soit sous Windows, Mac OS ou Linux. L'installateur est à lui seul extrêmement appréciable tant il est simple et convivial. Rien à reprocher là-dessus. L'installation des applications d'exemple AIR fonctionne sans l'ombre d'un problème avec là encore une très grande convivialité. Le développement de nouvelles applications AIR est particulièrement aisé avec Flex Builder et est complètement identique à Windows. On obtient ainsi rapidement un résultat probant.



# Créer une extension Firefox avec XUL

Firefox est reconnu pour sa grande modularité et la souplesse de ses interfaces. De nombreux développeurs proposent leurs créations et on peut notamment trouver une grande collection d'extensions et de thèmes sur le site officiel de Mozilla. L'objectif de cet article est de vous faire découvrir XUL et ses outils. Nous allons passer rapidement à la pratique en développant une extension Firefox et en abordant la création d'un thème.



APPLICATION : Web 2.0

LANGAGE : XUL

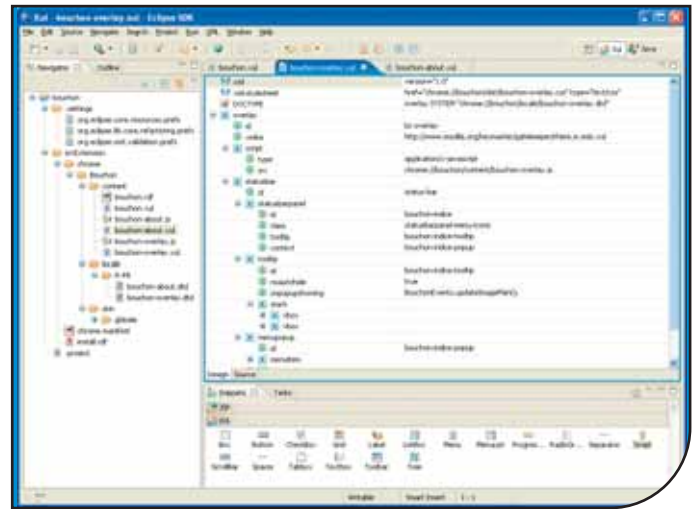
SOURCE : NON

XUL est un langage descriptif qui permet le développement d'interfaces graphiques. Il a été conçu par Mozilla pour l'ensemble

de ses produits et notamment Firefox et Thunderbird. Le moteur utilisé pour implémenter cette technologie est le même que pour la visualisation de pages web et l'exécution de code JavaScript : Gecko. Rien de plus normal quand on sait que XUL n'est autre qu'une extension des standards du web. Ainsi, développer une application XUL revient à utiliser des feuilles de styles CSS, du code JavaScript, l'API de traitement DOM et le méta-langage XML pour l'interface graphique. Il est donc relativement simple à appréhender pour les développeurs web et les designers d'autant qu'une documentation importante et une communauté active encadrent le projet. Le Mozilla Developer Center en est le point central. Contrairement au XHTML qui est un standard du W3C que tout le monde peut implémenter et utiliser librement, XUL n'a été décrit dans aucune spécification formelle. Pour cette raison, il ne dispose pas non plus de moteur de rendu interopérable : seul Gecko peut être utilisé. Bien que Gecko soit open source et que Mozilla offre un outil permettant d'exécuter les applications XUL en stand alone (XULRunner), XUL reste très dépendant de technologies non standard qui peuvent évoluer régulièrement, comme le montre Firefox 3. Malgré ces limitations apparentes d'interopérabilité, XUL est conçu pour être indépendant des plates-formes. Cela garantit la portabilité des applications sur l'ensemble des systèmes d'exploitation pour lesquels Firefox est disponible. Si l'on considère le large éventail de plates-formes qui supportent actuellement Mozilla, cela peut être une des caractéristiques intéressantes de XUL en tant que technologie pour développer des applications.

## LES COMPOSANTS D'UNE APPLICATION XUL

Une application XUL est organisée en trois parties principales : le contenu, le thème, les fichiers de localisation. Au niveau du contenu, on retrouve le document XUL qui définit l'interface utilisateur. Le thème regroupe quant à lui la feuille de style et les images. C'est lui qui définit l'apparence de l'application. La localisation repose sur plusieurs fichiers contenant des chaînes de caractères. Il s'agit d'un mécanisme permettant de faciliter la localisation et ainsi de proposer facilement le choix entre plusieurs langues à l'utilisateur. Ce choix est réalisé de manière transparente par Gecko. Par exemple, pour un Firefox en version française, les modules seront automatiquement en français. XUL définit de nombreux éléments, nous ne pouvons pas ici nous permettre d'en dresser une liste exhaustive, cependant, voici ceux que l'on retrouve le plus souvent et qui ont la plus grande importance :



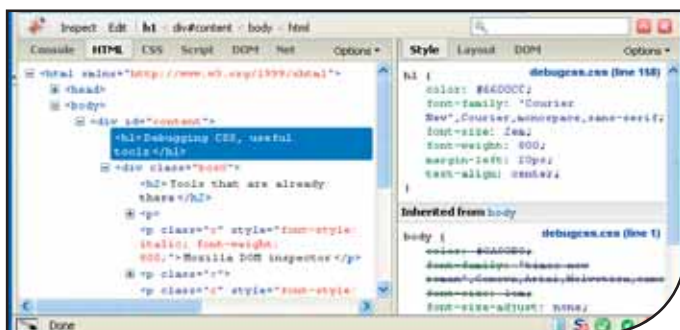
- Éléments de haut niveau (fenêtres, page, wizard...)
- Widgets (libellé, bouton, champ, menu...)
- Agencement (boîte, grille, pile, tas...)
- Événements et scripts
- Sources de données
- Objets utilitaires (navigateur, iframe...)

Il n'y a donc rien de nouveau au niveau des objets utilisés. Les développeurs Swing, Qt, GTK ou encore WxWidget ne devraient pas être perdus. Ici, la gestion des événements suit la logique de JavaScript. Ainsi, on spécifie la fonction à appeler pour chaque événement que l'on souhaite intercepter (onclick...). Les sources de données ne peuvent être que des nœuds XML que l'on exploite avec DOM. Les composants d'agencements permettent d'organiser le positionnement des composants avec une ergonomie et une maintenabilité maximale. Voici une petite application d'exemple qui affiche une boîte de dialogue avec trois boutons. On remarque que l'adoption du méta langage XML facilite beaucoup la compréhension du code grâce sa syntaxe descriptive.

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://global/skin/" type="text/css"?>

<window id="vbox exemple" title="Exemple"
xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">

<vbox>
  <label value="Email" control="email"/>
  <textbox id="email"/>
  <button id="ok" label="OK" onclick="Javascript:submit()" />
</vbox>
```



```
</window>
```

## LE MOTEUR GECKO

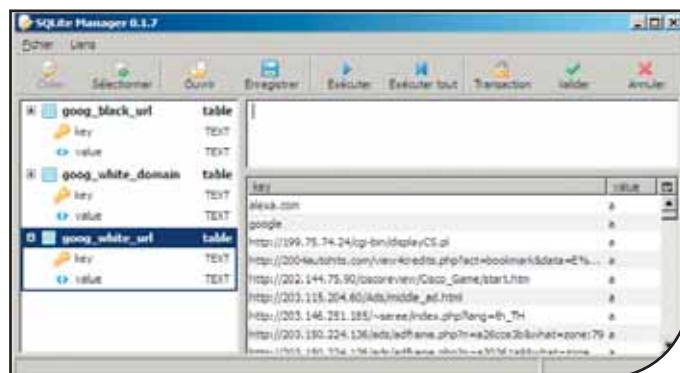
Gecko est le moteur de rendu utilisé par Firefox et par Thunderbird. On le retrouve également utilisé au sein des applications Flock, Camino, et SeaMonkey. Mais quel est l'objectif de Gecko ? La fondation Mozilla l'a conçu afin de supporter l'ensemble des standards du web. Ainsi, il permet l'affichage de contenu HTML, la prise en charge des styles CSS mais également un moteur d'exécution de JavaScript, le traitement de documents XML (DOM, XSLT, XPath) et le rendu d'images vectorielles avec le support du format SVG. Il se positionne ainsi en concurrent direct du moteur de rendu d'Apple, WebKit, qui est utilisé, entre autres par Safari et KDE 4. WebKit est également open source. Gecko offre une API très riche aux développeurs, ce qui le rend efficace dans de nombreux cas de figures : applications intégrant Internet, créer son propre navigateur, présentation de contenu, ou encore client / serveur. Gecko est cross-pla-teforme et exécutable sur de nombreux systèmes d'exploitation : Linux, \*BSD, Mac OS X, Windows, Solaris, AIX...

## ENVIRONNEMENT DE DÉVELOPPEMENT

La mise en place d'un environnement de développement est une étape importante lorsque l'on débute sur une nouvelle technologie. L'adoption d'un nouveau langage passe inévitablement par l'adoption de nouveaux outils. XUL n'échappe pas à la règle. Cependant n'oublions pas qu'il repose sur les standards W3C, ce qui facilite grandement la tâche. Ainsi pour éditer vos sources, vous pouvez utiliser n'importe quel éditeur texte (emacs, notepad++...). Cependant certains préfèrent un environnement complet pour développer. Il en existe deux qui sortent du lot : l'IDE Xul Dev ou Eclipse XUL. Dans notre cas d'étude, nous allons créer une extension et un thème pour Firefox. Nous pouvons donc installer des debuggers qui nous faciliteront le travail en cas de soucis. Je recommande notamment Firebug, et DOM inspector. Ce dernier s'installe avec Firefox. Il s'agit d'une option à cocher lors de l'installation avancée. Enfin, l'extension qui

### Chrome, c'est quoi ?

On entend parfois parler de Chrome, ce terme définit tout simplement le cadre et les widgets conteneur d'une fenêtre XUL. Ainsi, l'utilisation d'une URL Chrome permet d'afficher un fichier XUL directement dans la page. Par exemple, vous pouvez visualiser la fenêtre d'à propos, directement par le lien suivant : <chrome://browser/content/aboutDialog.xul>. Ce mécanisme est utilisé pour toutes les fenêtres et pop-up.



sera certainement la plus utile et qui porte bien son nom : Extension Developer's Extension. Celle-ci offre un générateur de package XPI et JAR, un éditeur de fichier d'installation, ou encore un éditeur pour XUL et XHTML en temps réel.

## CRÉONS NOTRE EXTENSION

L'extension d'exemple que nous allons créer va ajouter une entrée au menu option. Lorsque l'on cliquera dessus, elle affichera une fenêtre. Pour commencer, nous allons créer la structure de l'extension. Celle-ci est constituée d'un ensemble de fichiers, dont voici l'organisation générale : `helloprogrammez.xpi` :

```
_ content
_ hello.xul
_ overlay.js
_ overlay.xul
_ about.xul
_ local
_ en-US
  ■ hello.dtd
  ■ overlay.dtd
_ fr-FR
  ■ hello.dtd
  ■ overlay.dtd
_ skin
_ overlay.css
_ mainicon.png
_ install.rdf
```

Le fichier xpi, n'est en fait qu'un fichier zip sur lequel on aura renommé l'extension. Vous pouvez créer le dossier racine, nous allons éditer les fichiers. L'interface graphique est définie par le fichier `overlay.xul`. On définit ici l'interface utilisateur ainsi que les déclencheurs d'événements. On remarque que le contenu du label est en réalité une clef qui sera utilisé par Gecko pour la localisation du module. Nous allons définir les chaînes de caractères plus loin.

```
<?xml version="1.0"?>
<?xml-stylesheet href="chrome://HelloProgrammez/skin/overlay.
css" type="text/css"?>
<!DOCTYPE overlay SYSTEM "chrome://HelloProgrammez/locale/
overlay.dtd">
<overlay id="HelloProgrammez-overlay"
  xmlns="http://www.mozilla.org/keymaster/gatekeeper/
there.is.only.xul">
  <script src="overlay.js"/>

  <menupopup id="menu_ToolsPopup">
```

```
<menuitem id="HelloProgrammez-hello" label="&helloworld;"
          oncommand="HelloProgrammez.onMenuItemCommand
(event);"/>
</menupopup>
</overlay>
```

Dans notre cas, la logique de l'extension est simpliste mais suffisante pour appréhender le fonctionnement des événements. Le code qui correspond à la fenêtre précédente est *overlay.js*. Il s'agit d'un découpage arbitraire qui a l'avantage de structurer l'application :

```
var HelloProgrammez = {
  onLoad: function() {
    // initialization code
    this.initialized = true;
  },

  onMenuItemCommand: function() {
    window.open("chrome://helloprogrammez/content/hello.xul",
    "", "chrome");
  }
};

window.addEventListener("load", function(e) { HelloProgrammez
.onLoad(e); }, false);
```

Le fichier *hello.xul* définit la fenêtre qui est affichée par l'action de cliquer sur le menu :

```
<window xmlns="http://www.mozilla.org/keymaster/gatekeeper/
there.is.only.xul"
        title="&title.label;">

<hbox align="center">
  <description flex="1">&separate.label;</description>
  <button label="&close.label;" oncommand="close();"/>
</hbox>

</window>
```

Ensuite, la localisation de l'extension est implémentée dans le dossier local. Ici, on retrouve un dossier par langue. On peut en implémenter autant que souhaité. Dans ces dossiers correspond un fichier de propriété par interface. Voici le contenu de *hello.dtd* :

```
<!ENTITY title.label "Hello Programmez">
<!ENTITY separate.label "Hello Programmez!">
<!ENTITY close.label "Close">
```

Enfin, il ne reste plus qu'à définir le style de notre extension. Pour cela, nous allons ajouter une image *mainicon.png* dans le dossier *skin*. Nous allons également éditer la feuille de style *overlay.css*. Nous ajoutons une règle pour que notre entrée de menu soit bleue :

```
menuitem#helloprogrammez-hello {
  color: blue !important;
}
```

Nous venons de faire le tour de tous les aspects de l'implémentation d'un greffon. Désormais, il ne reste plus qu'à créer le fichier d'installation qui porte le nom suivant : *install.rdf*. Vous pouvez également

utiliser le plug-in de développement qui remplit parfaitement son rôle.

```
<?xml version="1.0"?>
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:em="http://www.mozilla.org/2004/em-rdf#">

  <Description about="urn:mozilla:install-manifest">

    <em:id>helloprogrammez@programmez.com</em:id>
    <em:name>Hello Programmez</em:name>
    <em:version>1.0</em:version>
    <em:description>Exemple d'extension</em:description>
    <em:creator>Loïc Guillois</em:creator>
    <!-- optional items -->
    <em:contributor>A person who helped you</em:contributor>
    <em:homepageURL>http://www.programmez.com</em:homepageURL>
    <!--em:optionsURL>chrome://helloprogrammez /content/settings
.xul</em:optionsURL>
    <em:aboutURL>chrome://helloprogrammez/content/about.xul</
em:aboutURL>
    <em:iconURL>chrome://helloprogrammez/skin/mainicon.png</em:
iconURL>
    <em:updateURL>http://sampleextension.mozdev.org/update.rdf
</em:updateURL-->

    <!-- Firefox -->
    <em:targetApplication>
      <Description>
        <em:id>{ec8030f7-c20a-464f-9b0e-l3a3a9e97384}</em:id>
        <em:minVersion>1.5</em:minVersion>
        <em:maxVersion>2.0.*</em:maxVersion>
      </Description>
    </em:targetApplication>

  </Description>

</RDF>
```

## CRÉER UN THÈME

Créer un thème pour Firefox est relativement simple. Cela ne demande que quelques compétences en style (CSS), probablement un peu XBL. Ce dernier est un langage complémentaire à XUL qui permet de décrire les propriétés des interfaces utilisateurs. L'organisation du package est aussi simple qu'une extension. Si vous souhaitez vous lancer dans l'aventure, le plus simple et le plus efficace est de récupérer un thème existant, le site officiel de Mozilla en regorge. Vous le customisez en changeant les éléments graphiques par les vôtres. Vous modifierez également les règles CSS.

## CONCLUSION

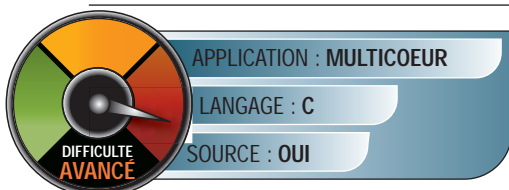
Nous avons vu ici les possibilités offertes par Gecko. On peut travailler l'interface de Firefox à volonté. Il est possible d'ajouter des extensions ou de créer un nouveau thème. Une flexibilité qui ne pénalise ni les performances ni la sécurité. La concurrence est assez distante et il n'y a vraiment que WebKit qui peut apporter des fonctionnalités équivalentes bien que moins puissant sur la mise en place d'extensions et de thèmes.

■ Loïc Guillois



# OpenMP et les processeurs multi-coeurs

Quoi de plus tentant pour un programmeur que d'exploiter les possibilités de sa nouvelle machine multi-cœur flambant neuve ? Pour cela **OpenMP propose une solution relativement aisée et portable.**

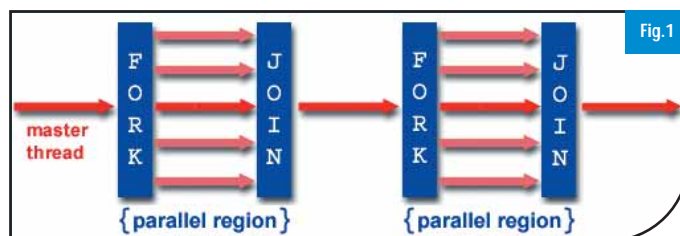


La programmation parallèle est depuis longtemps largement répandue. La plupart des interfaces l'explo-

itent pour éviter que l'utilisateur n'ait l'impression qu'une application reste bloquée lorsqu'elle effectue un travail important. Nous parlons là des "vieilles" applications multithreadées, mais fonctionnant sur un seul processeur, c'est-à-dire sans gain de performances globales, ou même plutôt avec une perte de performances globales, puisque le système perd du temps et des ressources pour la gestion et l'ordonnancement des threads. Aujourd'hui le multi-cœur se généralise et donne une nouvelle dimension à la programmation parallèle. Il n'est alors plus question de se préoccuper essentiellement du confort de l'utilisateur au détriment des performances, mais bien d'exploiter les possibilités du matériel pour un gain de performances brutes. Il est évident qu'une application qui ne tourne que sur un seul cœur d'un CPU multi-cœur sera même moins performante que la même application tournant sur un CPU à l'ancienne, sans doute plus rapidement cadencé. En revanche si le flux d'exécution de l'application se répartit sur tous les cœurs, alors le gain de performance sera directement multiplié par le nombre de cœurs utilisés. La question est de savoir comment répartir ce flux d'exécution. OpenMP est une excellente réponse.

## 1 PRÉSENTATION SUCCINCTE D'OpenMP

OpenMP est une spécification définie par des industriels, ce qui lui donne un caractère indéniablement pragmatique. Ces spécifications ciblent trois langages : Fortran, C et C++. Elles sont aujourd'hui largement implémentées, ce qui permet d'écrire un code facilement portable, quand il n'est pas directement portable. L'implémentation de ces spécifications aboutit à une librairie d'exécution (runtime). Mais l'originalité de la chose est que si l'on y invoque des fonctions comme avec toute librairie, la déclaration des blocs devant être exécutés en parallèle n'est pas codée avec le langage lui-même mais avec un jeu de directives pouvant être affinées par des clauses. Ces directives sont incluses directement dans le code source. Concrètement, en C et C++ cela revient à insérer des pragmas dans le source et c'est ensuite le compilateur qui se charge de générer le code natif parallèle qui doit en découler. Le modèle de programmation OpenMP est dit "Fork and Join", ce que l'on peut traduire par éclatement regroupement. Pour OpenMP toute application compte au moins un thread qui est tout simplement le programme dans son exécution séquentielle. Quand le code arrive à un point où se situe une section parallélisée, d'autres threads sont créés. Le premier thread, dit master, a la maîtrise sur l'équipe de threads dont il fait lui-même partie et dont il est en quelque sorte le capitaine. Lorsque l'exécution de la section parallélisée est terminée, les threads sont détruits, à l'ex-



ception du master qui reprend une exécution séquentielle du programme comme illustré sur la figure 1. A ce moment, le résultat du travail des autres membres de l'équipe peut être collecté si besoin. Pour résumer, programmer avec OpenMP, c'est écrire des programmes multithreads, mais sans devoir se préoccuper de la fastidieuse mise en oeuvre des threads natifs, mise en oeuvre non seulement fastidieuse et difficile, mais de surcroît spécifique à chaque système, voire à chaque librairie de thread.

## 2 LES OUTILS

Nous allons écrire quelques exemples montrant tout l'intérêt d'OpenMP sur les architectures multi-cœurs. Pour cela, nous nous basons sur les spécifications 2.0 d'OpenMP. Le lecteur désireux d'approfondir ses connaissances peut consulter le document à [www.openmp.org](http://www.openmp.org). Les exemples ont été testés sur une machine équipée d'un CPU Intel Q6600 à quatre cœurs. La machine est en dual boot. Linux avec un kernel 2.6 d'une part, Windows Vista d'autre part. Le code est écrit en C car sous Linux nous avons utilisé une implémentation pour le langage C : OMPI qu'il est possible de télécharger à <http://www.cs.uoi.gr/~ompi/>. Pour installer celle-ci sur votre système, vous devez passer par la traditionnelle séquence de commandes :

```
./configure
make
make install
```

OMPI se présente comme un frontal du compilateur de votre choix et son runtime est par défaut basé sur la librairie POSIX pthread. Le compilateur d'OMPI traite les directives OpenMP qu'il trouve dans votre code pour l'enrichir de toute la plomberie de création des threads avec *pthread*. Ce code enrichi est ensuite compilé par défaut par *gcc*. Installer et utiliser OMPI ne présente aucune difficulté. Cependant prenez le temps de lire en détail la procédure d'installation. Vous y verrez comment il est (facilement) possible de demander l'utilisation d'un compilateur autre que *gcc*, ou bien d'une autre librairie de threads. Pour cet article nous avons utilisé toutes les options par défaut. Compiler et faire l'édition de liens est immédiat. Par exemple :

```
ompiicc -o helloworld helloworld.c
```



Fig.2

Si vous êtes curieux de lire le code intermédiaire généré par le compilateur *ompicc*, ajoutez le commutateur *-k* à la ligne de commande précédente. Mentionnons en passant que nous avons aussi essayé notre code avec le compilateur d'Intel. La version libre de ce compilateur vient avec une implémentation d'OpenMP. Les résultats obtenus sont identiques. Avec Intel++, du code OpenMP se compile ainsi :

```
icc -openmp -o helloworld helloworld.c
```

Sous Windows Vista nous avons utilisé Visual Studio 2008. Avec cet outil, tout comme avec Visual Studio 2005, OpenMP n'est hélas disponible qu'à partir de l'édition professionnelle. Avec l'un comme l'autre outil, la ligne de commande passée au compilateur doit comporter le commutateur */openmp*. Ce commutateur peut être défini depuis l'IDE, comme illustré sur la figure 2. Sous Visual Studio 2008, l'édition de liens avec le runtime OpenMP sera effectuée automatiquement. Sous Visual Studio 2005 vous devrez demander cette édition de liens explicitement, par exemple en ajoutant une directive dans votre code comme ceci :

```
#pragma comment(lib, "vcomp.lib") // ou vcompd.lib si débogage
```

En outre, vous devrez veiller à ce que les bibliothèques *vcomp.dll* et *vcompd.dll* soient pointées par le PATH du système. Enfin si vous n'avez pas Visual Studio, vous pourrez quand même vous essayer à OpenMP en installant Cygwin sur votre système, ce qui vous permettra alors de travailler avec OPMi.

### 3 POUR SE FAIRE LA MAIN

OpenMP dispose de fonctions pour connaître le nombre de threads qu'elle gère par défaut, ou pour positionner ce nombre. Faisons un petit essai:

```
// nb_threads.c

#include <stdio.h>
#include <omp.h>

const int nb_threads = 1000000;

// Eventuellement, avec Visual Studio
#pragma comment(lib, "vcomp.lib")
```

```
int main(void)
{
    int nb_procs = omp_get_num_threads();

    printf("Nb de processeurs: %d\n", nb_procs);
    printf("Nb maxi de threads: %d\n", omp_get_max_threads());
    omp_set_num_threads(nb_threads);
    printf("Nb maxi de threads: %d\n", omp_get_max_threads());
    return 0;
}
```

Notre code, comme tout code OpenMP commence par inclure l'entête *omp.h*. Ensuite, nous invoquons *omp\_get\_num\_threads* pour connaître combien de threads peuvent être instanciés par défaut. Sur notre machine, toutes les implémentations d'OpenMP ont retourné la valeur 4, soit le nombre de coeurs du CPU. Ensuite, nous positionnons le nombre de threads à un million, puis nous appelons à nouveau *omp\_get\_num\_threads* pour savoir où nous en sommes. Ici les résultats diffèrent selon le système d'exploitation. Sous Windows Vista c'est la valeur 64 qui est retournée. Cette valeur est depuis toujours le nombre de threads maximum par application sous Windows. Que nous disposions d'un CPU multi-cœur n'y change absolument rien. Sous Linux, en revanche, nous disposons maintenant d'un million de threads potentiels :) Bien sûr cette valeur est totalement fantaisiste. Le runtime consommerait plus de temps à ordonnancer les threads qu'à les exécuter. En revanche il est intéressant de pouvoir positionner le nombre de threads maximum au gré de l'exécution d'une application. En effet une application peut avoir besoin de 1, 2, ou 3 threads à un moment donné de son exécution. Il faut toujours bien garder à l'esprit que si l'on ne dit rien à OpenMP celle-ci, lorsqu'elle rencontre une section parallélisée, crée le nombre de threads maximum défini à l'instant. Ce nombre doit donc toujours être en adéquation avec la logique de l'application.

## 4 UN HELLO WORLD! PARALLÉLISÉ

La tradition est la tradition. Voyons donc comment écrire un Hello World! parallélisé:

```
// HelloWorld.c

#include <stdio.h>
#include <omp.h>

#pragma comment(lib, "vcompd.lib")

int main(void)
{
    int id;

    //omp_set_num_threads(3);

    #pragma omp parallel private(id)
    {
        id = omp_get_thread_num();
        printf("Hello depuis le thread %d\n", id);
    }
}
```

```
return 0;
}
```

Ce rudimentaire programme est riche d'enseignements. Au tout début nous déclarons un entier, qui sera un identifiant de thread. Remarquons que l'on ne déclare qu'un seul entier, alors que l'on sait d'avance qu'il y aura plusieurs threads. Ensuite nous avons en commentaire un appel à *omp\_set\_num\_threads* pour positionner le nombre de thread à 3. Tant que cet appel reste en commentaire, le nombre de threads sera 4 ainsi que nous l'avons vu dans l'exemple précédent. Vous pourrez ici essayer d'utiliser cet appel pour positionner à trois ou moins, ou même à une valeur supérieure à 4 et constater ce qui se passe. Nous rencontrons ensuite notre première directive OpenMP: *parallel*. Cette directive est signalée au compilateur par *#pragma omp*. Cette directive demande que le flux d'exécution, qui était jusqu'ici monothread, soit éclaté en nombre de threads prédéfinis en amont. Le thread initial fait partie du nombre et porte le nom de "master thread". La section de code ainsi parallélisée se situe entre accolades. On sait qu'en C/C++ une paire d'accolades peut toujours délimiter un bloc de code. Au stade où nous en sommes, cela veut dire que le code entre accolades est exécuté *n* fois à l'identique. Ceci présente-t-il un intérêt pratique ? Aucun. Il est évident que chaque thread doit effectuer une tâche particulière dans une application multithreadée. C'est au programmeur de tester dans quel thread il se situe et d'agir en conséquence. Pour cela la fonction *omp\_get\_thread\_num* invoquée dans une section parallélisée retourne un numéro identifiant l'instance du thread depuis lequel la fonction a été appelée. Ici nous rangeons cette valeur dans la variable *id* déclarée avant la section parallélisée. Y-a-t-il conflit entre les threads qui accèdent tous à cette variable ? Non. En effet dans le code, la directive parallèle est enrichie d'une clause: *private*. Ici *private(id)* a pour effet de dupliquer la variable *id* pour chaque thread, chaque instance de cette variable, comme le terme *private* le suggère, étant visible uniquement dans le contexte de son thread. Nous comprenons maintenant comme notre code fonctionne et parvient à nous afficher fièrement :

```
Hello depuis le thread 0
Hello depuis le thread 1
Hello depuis le thread 2
Hello depuis le thread 3
```

Sans que nous en ayons encore conscience, nous venons d'être salués par chacun des coeurs de notre CPU :) Nous y reviendrons. Avant cela notre code mérite encore quelques remarques. D'abord, il est bon de savoir que le "master thread" ou thread principal de l'application, porte toujours l'identifiant 0. Ensuite, dans une application OpenMP, il est vivement recommandé d'utiliser les fonctions d'entrées/sorties C (*puts*, *printf*, ...) même dans du code C++, car la nature des flux de C++ peut provoquer des sorties cahotiques sur certaines implémentations. Enfin, il est bon de savoir que l'accolade fermante de notre section parallélisée constitue une barrière implicite. Concrètement, cela veut dire que le master thread attend ici que tous les threads (lui inclus) de la section aient terminé pour reprendre le flux d'exécution monothread. Bien entendu, OpenMP propose d'autres moyens raffinés pour la synchronisation. Nous renvoyons le lecteur à la documentation. Positionner le nombre de threads est tellement fréquent dans une application OpenMP, qu'il

existe une clause spécialement pour cela, *num\_threads*, et qu'en général, on l'utilise de préférence à *omp\_set\_num\_threads* :

```
// HelloWorld2.c

#include <stdio.h>
#include <omp.h>

#pragma comment(lib, "vcompd.lib")

int main(void)
{
    int id;
    #pragma omp parallel num_threads(3) private(id)
    {
        id = omp_get_thread_num();
        printf("Hello depuis le thread %d\n", id);
    }
    return 0;
}
```

## 5 UN GAIN DE PERFORMANCE CLAIR

OpenMP s'occupe de paralléliser, mais c'est finalement l'architecture matérielle et le système d'exploitation qui seront responsables des performances. Nos exemples précédents auraient très bien fonctionné sur une machine à l'ancienne, dotée d'un seul CPU. Dans ce cas, c'est ce CPU qui exécute tous les threads. Si OpenMP présente l'intérêt de faire des calculs en parallèle, la gestion des threads provoquerait ici une perte de performances. Sur une machine multi-core, OpenMP prend une autre dimension, car tant que le nombre de threads ne dépasse pas le nombre de coeurs, chaque thread tourne sur un coeur. Le gain de performance est alors directement fonction du nombre de coeurs, la gestion de la répartition de charge apparaissant négligeable dans ce contexte. A condition bien entendu que le système d'exploitation sache tirer parti du matériel. C'est parfaitement le cas de Windows Vista sur lequel nous avons fait nos essais. C'est également parfaitement le cas avec un Linux doté d'un noyau récent, un 2.6.23 en ce qui nous concerne, et à condition que ce noyau soit compilé pour gérer les multi-coeurs. Pour cela cochez :

```
Processor type and features -->
[*] Multicore scheduler support
```

dans le menu de configuration du noyau, comme illustré [Fig.3]. Voici maintenant un programme sans autre intérêt que de dévorer du temps CPU :

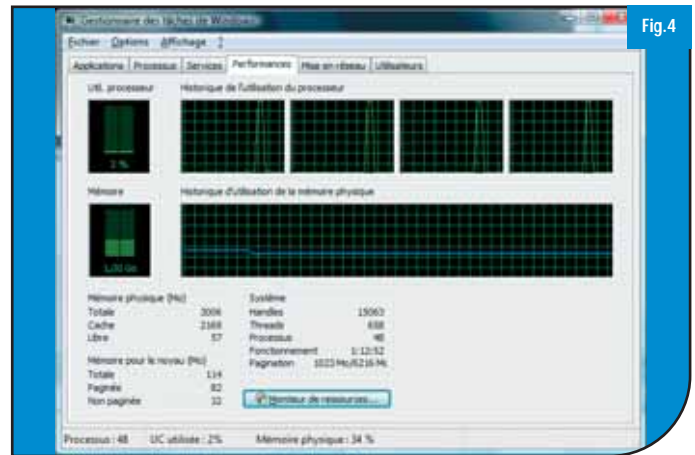
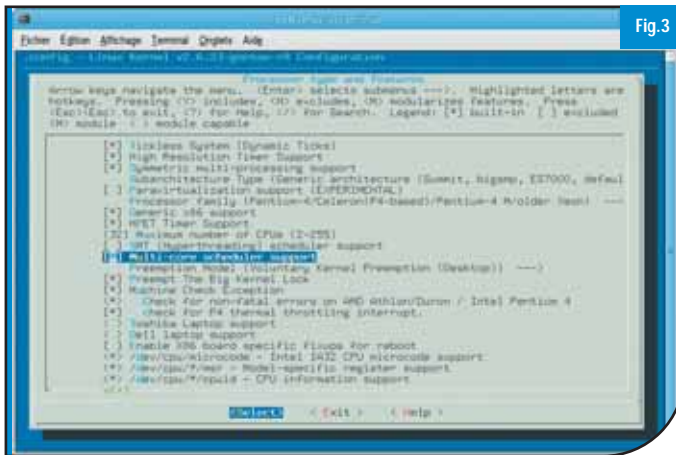
```
// no_thread.c

#include <stdio.h>
#include <time.h>

const long nb_loop = 400;

void go(void)
{
```





```
time_t debut, fin, temp;
int i,j,k;

time(&debut);

for(i=0; i<nb_loop; i++)
{
    for(j=0; j<nb_loop; j++)
        for(k=0; k<nb_loop; k++)
            time(&temp);
}

time(&fin);
printf("Termine\n");
printf("Temps ecoule: %ld\n", fin-debut);
}

int main(void)
{
    go();
    return 0;
}
```

Sur notre machine dont c'était le seul travail au moment des essais, ce code a demandé 10 secondes pour s'exécuter sous Linux et 4 secondes sous Windows Vista. Nous ne nous posons pas aujourd'hui la question de savoir pourquoi il y a une telle différence entre les deux systèmes. En revanche, nous parallélisons le code, afin que la boucle soit exécutée quatre fois :

```
// boucle.c

#include <stdio.h>
#include <time.h>
#include <omp.h>

const long nb_loop = 400;

void go(void)
{
    time_t debut, fin, temp;
    int id;
```

```
time(&debut);
#pragma omp parallel private(id, temp)
{
    int i,j,k;
    id = omp_get_thread_num();

    // parce que ompicc ne comprend pas for(int i=0....
    for(i=0; i<nb_loop; i++)
    {
        for(j=0; j<nb_loop; j++)
            for(k=0; k<nb_loop; k++)
                time(&temp);
    }
    printf("Thread %d termine\n", id);
}
time(&fin);
printf("Tout termine -- barriere implicite atteinte\n");
printf("Temps ecoule: %ld\n", fin-debut);
}

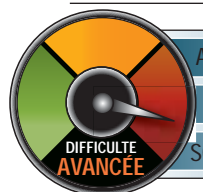
#pragma comment(lib, "vcomp.lib")

int main(void)
{
    //omp_set_num_threads(4);
    go();
    return 0;
}
```

Que constatons nous ? Le temps d'exécution reste à 10 et 4 secondes sous Linux et Windows respectivement. Autrement dit, l'exécution des 3 threads supplémentaires à un coût absolument négligeable et le système répartit bien la charge sur chacun des coeurs. Nous pouvons d'ailleurs facilement le vérifier avec le gestionnaire de tâches de Windows. [Fig.4] En conclusion, nous dirons qu'OpenMP présente un double attrait : faciliter l'écriture de code parallèle tout en profitant de la puissance des microprocesseurs multi-coeurs modernes.

# Observez la charge de votre processeur multi-cœur sous Windows

L'API d'instrumentalisation de Windows, ou WMI, permet de connaître dans les moindres détails ce qui se passe au sein d'une machine. Nous voyons aujourd'hui comment observer la charge de chaque cœur d'un microprocesseur, avec C++ et C#.



APPLICATION : PROCESSEUR

LANGAGE : C++ / C#

SOURCE : OUI

En guise de visite guidée dans les entrailles de Windows, nous vous proposons aujourd'hui de découvrir comment

observer la charge de chaque cœur d'un processeur multi-cœur, et, pourquoi pas, comment être notifié de manière asynchrone du fait qu'un cœur atteint un certain seuil de charge. Pour l'occasion nous n'allons pas sortir un assembleur du placard, mais travailler avec une API Windows extrêmement puissante, la WMI. La Windows Management Instrumentation, ou WMI pour les intimes, est l'implémentation de spécifications définies par un groupe d'industriels: La Distributed Management Task Force. L'idée est de pouvoir administrer n'importe quel système, localement ou à distance, au moyen d'une interface normalisée. Grosso modo, ces spécifications consistent en deux couches. La première couche est la CIM. Dans cette couche sont définies des classes qui présentent une très forte analogie avec les classes de la programmation objet. Ainsi une classe encapsule des données relatives au système, éventuellement accompagnées de méthodes. Une classe peut dériver d'une autre. En revanche, à la différence de la programmation objet, le langage utilisé ne peut pas instancier directement une classe, comme on le fait, par exemple en C++ C# ou Java, en invoquant l'opérateur new. A la place, on doit appeler des méthodes de classes qui présentent une analogie avec les méthodes statiques de la programmation objet. La deuxième couche de spécifications est la WBEM. Elle constitue une voie d'accès standard aux classes CIM et dérivées. L'implémentation sous Windows de ces spécifications est la WMI. Si nous nous penchons aujourd'hui sur Windows uniquement, le lecteur pourra être intéressé par une implémentation Open Source de WBEM pour Linux. Celle-ci se trouve à <http://openwbem.org/>

## 1 UNE VISION SIMPLE DE LA WMI WINDOWS

La WMI est une API très complexe, mais bien documentée par la documentation MSDN. On y trouve d'un côté des jeux de classes qui étendent les classes CIM. Prenons par exemple la classe Win32\_Process. Celle-ci dérive de la classe CIM CIM\_Process en lui donnant des propriétés supplémentaires et en la dotant de méthodes, telles que *Create* qui permet de l'instancier, ce qui aura alors pour effet de lancer un processus. De l'autre côté vient l'implémentation de la WBEM. Sous Windows, cette implémentation est faite en COM, architecture qui convient parfaitement au propos. La programmation COM n'est pas réputée facile, toutefois l'encapsulation sous .Net simplifie considérablement les choses, et, en ce qui

concerne C++, on s'en tire facilement avec quelques classes auxiliaires. Tout notre code, C# ou C++, a été écrit avec Visual Studio 2008, mais devrait pouvoir être exploité sans difficulté avec l'outil de votre choix. Le code a été testé sous Windows Vista, mais devrait fonctionner aussi sous Windows XP. Aucune garantie n'est donnée en ce qui concerne les systèmes Windows antérieurs. Sous Vista le code doit être lancé en tant qu'administrateur. Pour les essais, le plus simple est de lancer Visual Studio ainsi, les applications lancées ensuite par lui héritent automatiquement des droits.

## 2 POUR SE FAIRE LA MAIN

Commençons par écrire un petit morceau de code C# voulant énumérer les processeurs présents sur notre machine. Celle-ci est équipée d'un Intel Q6600 à quatre cœurs non hyperthreadé. Nous nous attendons donc à ce que le code nous affiche fièrement la présence de 4 processeurs.

```
using System;
using System.Management;

namespace CSharpEnumProcessors
{
    class Program
    {
        static void Main(string[] args)
        {
            ManagementClass mc =
                new ManagementClass("Win32_Processor");
            ManagementObjectCollection cpus =
                mc.GetInstances();
            foreach (ManagementObject cpu in cpus)
            {
                PropertyDataCollection cpuProperties
                    = cpu.Properties;
                foreach (PropertyData cpuProperty in cpuProperties)
                {
                    Console.WriteLine(cpuProperty.Name + " = ");
                    Console.WriteLine(cpuProperty.Value);
                }
            }
        }
    }
}
```

Avant tout, ce code ne peut être compilé que si l'assembly Sys-

tem.Management est ajouté aux références du projet. Les deux premières lignes de la méthode main font un travail formidable en masquant de nombreuses étapes que nous aurons à suivre avec C++. La classe .Net ManagementClass encapsule la notion de classe CIM. La méthode GetInstances de ManagementClass retourne une collection de toutes les instances de la classe CIM au sein du système. Chacune de ces instances est encapsulée dans une classe ManagementObjet. L'accès à la propriété Properties de la classe retourne une collection des propriétés de la classe CIM, qu'il ne reste plus qu'à afficher. Ce code n'est pas très compliqué, mais il ne se comporte pas conformément à nos attentes. Nous ne voyons listé qu'un seul processeur. Et selon le contexte, la propriété correspondant à la charge du processus sera peut-être vide. En fait, ce résultat, auquel ne nous nous attendions pas est parfaitement logique. Nous n'avons réellement qu'un seul processeur, mais à quatre coeurs. Ainsi voyons nous affiché :

```
NumberOfCores = 4
NumberOfLogicalProcessors = 4
```

Mais si dans la documentation MSDN nous voyons bien la classe Win32\_Processor, nous ne voyons pas de classe baptisée Win32\_Core, ni rien qui puisse y ressembler. Comment faire alors pour accéder à chaque coeur individuellement ?

### 3 CLASSES WIN32\_PERF ET DÉRIVÉES.

Ce que nous voulons faire n'est pas nous informer de la présence de matériel, mais bien nous enquérir au sujet des performances de celui-ci. Le cas est prévu dans les spécifications de la CIM qui définit la classe CIM\_StatisticalInformation pour cela. Microsoft a dérivé cette classe en un jeu de classes destinées au suivi des performances. Vient d'abord Win32\_Perf, puis de cette dernière dérivent deux branches de classes, dont les noms sont préfixés Win32\_PerfRawData pour les classes fournissant des données brutes et Win32\_PerfFormattedData pour les classes fournissant des données prêtes à l'emploi. Afin de ne pas accroître la charge de système, ces classes ne sont pas instanciées, donc pas actives, par défaut, ce qui explique que notre premier code peut présenter un affichage vide pour la charge du microprocesseur. En revanche, si vous relancez notre premier exemple après avoir essayé les autres, il est probable que la charge apparaisse à ce moment :) Si on examine ces deux branches de classes, nous y trouvons les classes Win32\_PerfRawData\_PerfOS\_Processor et Win32PerfFormattedData\_PerfOS\_Processor qui serviront notre propos. En effet, et comme leur nom ne l'indique pas, il y aura une instance de classe par processeur physique, mais aussi par processeur logique, c'est-à-dire pour nous, une instance par coeur. Nous allons travailler avec Win32PerfFormattedData\_PerfOS\_Processor et en afficher, pour chaque instance, les propriétés suivantes :

Nom	Signification
Name	Nom/rang du processeur ou du coeur
PercentIdleTime	Pourcentage de temps passé à ne rien faire.
PercentInterruptTime	Pourcentage de temps passé au traitement d'interruptions matérielles.
PercentPrivilegedTime	Pourcentage de temps passé dans le mode privilégié. Par exemple en exécutant le code d'un pilote.
PercentUserTime	Pourcentage de temps passé en mode utilisateur.
PercentProcessorTime	Pourcentage de temps processeurs calculé ainsi : 100% - PercentIdleTime.

## 4 WMI ET C++

Nous allons maintenant travailler en C++ (projet CppEnumCores sur le CD-Rom et/ou le site). C'est un peu plus difficile qu'en C#, mais on fait très exactement ce que l'on veut. En outre, cela permet d'avoir une bonne connaissance du fonctionnement des mécanismes. Lorsque nous reviendrons à C# à la fin de cet article, nous verrons que cette connaissance n'est pas superflue. Avec C++, nous devons nous attaquer à la manipulation d'interfaces COM. Comme nous l'avons souvent fait dans d'autres articles de programmation Windows, nous fournissons un jeu de classes destinées à nous aider dans notre travail, et notamment un pointeur intelligent qui se charge du comptage des références d'interfaces COM. Nous profitons ainsi au maximum de l'idiome C++ "Acquisition de ressources, pas initialisation". En cas d'erreur signalée par une méthode d'interface COM, nous levons une exception, ce qui a pour effet de libérer toutes les ressources. Par ailleurs nous supposons que le lecteur est un fidèle de Programmez! et que comme tel il est très familiarisé avec la programmation COM :) Au départ de l'application, La plomberie COM doit être initialisée. Notre code commence donc ainsi :

```
void DoIt()
{
    SmartCOMPtr<IWbemLocator> locator;
    SmartCOMPtr<IWbemServices> services;
    SmartCOMPtr<IWbemRefresher> refresher;
    SmartCOMPtr<IWbemConfigureRefresher> configure;
    SmartCOMPtr<IWbemHiPerfEnum> perfenum;

    SetCOMSecurity();
    locator = GetWbemLocator();
    services = GetWbemServices(locator);
    refresher = GetWbemRefresher();
    configure =
        GetWbemConfigureRefresher(refresher);
    perfenum = GetPerfEnum(services,
        configure,
        L"Win32_PerfFormattedData_PerfOS_Processor");

    for(int i = 0; i < 20; i++)
    {
        GetObjects(refresher, perfenum);
        cout << "-----" << endl << endl;
        ::Sleep(2000);
    }
}

int main()
{
    COMInitializerEx com_initializer_ex;
    try
    {
        DoIt();
    }
    catch(COMException ce)
    {
        std::cout << ce.raison() << std::endl;
        return EXIT_FAILURE;
    }
}
```



```

}

return 0;
}

```

Le constructeur de la classe auxiliaire COMInitializerEx se charge de l'initialisation de la plomberie et son destructeur de la fermeture. Le travail du programme est effectué dans la fonction *Dolt* dont l'appel est placé dans un bloc *try*. Nous allons examiner plus loin dans le détail ce qui est fait par *Dolt*. Un premier coup d'oeil montre qu'après une série d'initialisation, une boucle va, une vingtaine de fois, récupérer à la volée les instances des classes pour afficher leurs propriétés. Examinons d'abord le code qui précède cette boucle, et qui est de loin le plus délicat. Nous commençons par nous octroyer des droits généreux :

```

void SetCOMSecurity()
{
    HRESULT hr;

    hr = hr = ::CoInitializeSecurity(
        NULL,
        -1,
        NULL,
        NULL,
        RPC_C_AUTHN_LEVEL_NONE,
        RPC_C_IMP_LEVEL_IMPERSONATE,
        NULL, EOAC_NONE, 0);
    COMHelper::TestOk(hr);
}

```

Nous devons maintenant nous connecter à la WMI de notre machine et manipuler ces classes. Celles-ci résident dans un espace de noms. Par défaut sous Windows, celui-ci se nomme ROOT\CIMV2. Cette information peut être vérifiée dans la base de registre :

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WBEM\Scripting
```

Vérifier cette information et éventuellement modifier notre code en conséquence peut permettre de le faire fonctionner sur une machine d'entreprise dont la configuration serait particulière. Le lecteur à l'esprit curieux trouvera d'autres informations très intéressantes dans les clés sous WBEM. Tout cela est très bien, mais ne nous dit pas comment accéder concrètement à l'API. Cela se fait en deux temps. D'abord on doit obtenir l'interface COM IWbemLocator, dont le rôle est de localiser l'espace de nom et de fournir un pointeur sur l'interface IWbemServices. Cette dernière est la clé de voûte de la programmation WMI C++ car c'est à travers elle que s'obtiennent classes, instances de classes ou encore énumérateurs WMI. Commençons par obtenir l'instance de l'interface COM IWbemLocator :

```

#include <Wbemidl.h>

#pragma comment(lib, "wbemuuid.lib")

SmartCOMPtr<IWbemLocator> GetWbemLocator()
{

```

```

    HRESULT hr;
    SmartCOMPtr<IWbemLocator> locator;

    hr = ::CoCreateInstance(
        CLSID_WbemLocator,
        NULL,
        CLSCTX_INPROC_SERVER,
        IID_IWbemLocator,
        reinterpret_cast<void*>(&locator));
    COMHelper::TestOk(hr);
    return locator;
}

```

A ce stade, ce n'est que de la programmation COM classique. A partir de notre IWbemLocator, nous pouvons maintenant obtenir une instance de IWbemServices :

```

SmartCOMPtr<IWbemServices>
GetWbemServices(SmartCOMPtr<IWbemLocator> locator)
{
    HRESULT hr;
    SmartCOMPtr<IWbemServices> services;

    hr = locator->ConnectServer(
        L"ROOT\\CIMV2",
        NULL,
        NULL,
        0,
        NULL,
        0,
        0,
        &services);
    COMHelper::TestOk(hr);
    return services;
}

```

A ce point, il serait tentant d'essayer d'accéder aux instances de Win32\_PerfFormattedData\_PerfOS\_Processor, comme on le fera par exemple pour Win32\_Processor. Seulement cela ne donnerait rien. Il se trouve que les classes de performances sont traitées différemment des autres par l'API. Pour y accéder, on doit passer par l'intermédiaire d'une interface COM IWbemRefresher qui, comme son nom le suggère, va se charger de rafraîchir en temps réel les données contenues dans les instances de classes. Nous devons donc obtenir cette interface, qui est une interface COM comme une autre. Nous n'utilisons donc pas encore IWbemServices :

```

SmartCOMPtr<IWbemRefresher> GetWbemRefresher()
{
    HRESULT hr;
    SmartCOMPtr<IWbemRefresher> refresher;

    hr = ::CoCreateInstance(
        CLSID_WbemRefresher,
        NULL,
        CLSCTX_INPROC_SERVER,
        IID_IWbemRefresher,
        reinterpret_cast<void*>(&refresher));

```

```
COMHelper::TestOk(hr);
return refresher;
}
```

Qu'est ce que notre interface devra rafraîchir ? Elle n'en sait encore rien. Nous devons lui dire au moyen d'une interface de configuration qui est l'interface COM IWbemConfigureRefresher. Nous ne devons pas travailler avec une instance prise au hasard, mais avec l'instance qui est associée à l'instance de notre IWbemRefresher. C'est pour cela que nous passons par la méthode QueryInterface de cette dernière. Cette démarche est obligatoire :

```
SmartCOMPtr<IWbemConfigureRefresher>
GetWbemConfigureRefresher(SmartCOMPtr<IWbemRefresher> refresher)
{
    HRESULT hr;
    SmartCOMPtr<IWbemConfigureRefresher> configure;

    hr = refresher->QueryInterface(
        IID_IWbemConfigureRefresher,
        reinterpret_cast<void **>(&configure));
    COMHelper::TestOk(hr);
    return configure;
}
```

Il est maintenant temps de faire la configuration. Il s'agit ici de dire au rafraîchisseur quelles classes il doit traiter, et lui dire de placer les résultats dans une énumération COM, à laquelle nous accédons nous-mêmes par une instance de l'interface COM IWbemHiPerfEnum. Un pointeur sur cette instance nous est retourné à l'issue de l'opération ci-dessous :

```
// classe = "Win32_PerfFormattedData_PerfOS_Processor"
SmartCOMPtr<IWbemHiPerfEnum>
GetPerfEnum(
    SmartCOMPtr<IWbemServices> services,
    SmartCOMPtr<IWbemConfigureRefresher> configure,
    LPCWSTR classe)
{
    HRESULT hr;
    long ID;

    SmartCOMPtr<IWbemHiPerfEnum> perfenum;

    hr = configure->AddEnum(
        &(*services),
        classe,
        0,
        NULL,
        &perfenum,
        &ID);
    COMHelper::TestOk(hr);
    return perfenum;
}
```

On remarque ici que l'interface IWebService est enfin de la partie. On remarque encore que si l'on configure une instance de IWebRefresher celle-ci n'apparaît pas dans le code ci-dessus. N'oublions pas

que nous travaillons avec l'interface de configuration associée au rafraîchisseur en question, ce qui est suffisant, et justifie pourquoi il n'y a qu'une seule manière d'obtenir l'interface de configuration. Tout est prêt, il ne reste plus qu'à invoquer la fonction GetObjects de notre exemple à intervalle régulier, en lui passant le rafraîchisseur et l'interface d'énumération. En voici le code partiel. Nous avons retiré les éléments répétitifs :

```
void GetObjects(
    SmartCOMPtr<IWbemRefresher> refresher,
    SmartCOMPtr<IWbemHiPerfEnum> perfenum)
{
    HRESULT hr;
    DWORD nb_objects = 0;
    const DWORD size = 20;
    IWbemObjectAccess* objects[20];

    hr = refresher->Refresh(0L);
    COMHelper::TestOk(hr);

    hr = perfenum->GetObjects(0L,
        size,
        objects,
        &nb_objects);

    if (hr == WBEM_E_BUFFER_TOO_SMALL )
    {
        cout << "Tableau trop petit pour contenir les objets" << endl;
        cout << "Redimensionnez le tableau et recompilez le code :)"
            << endl;
        throw COMException(E_INVALIDARG);
    }

    CIMTYPE PercentIdleTimeType;
    CIMTYPE PercentInterruptTimeType;
    CIMTYPE PercentPrivilegedTimeType;
    CIMTYPE PercentUserTimeType;
    CIMTYPE PercentProcessorTimeType;

    long PercentIdleTimeHandle;
    long PercentInterruptTimeHandle;
    long PercentPrivilegedTimeHandle;
    long PercentUserTimeHandle;
    long PercentProcessorTimeHandle;

    hr = objects[0]->GetPropertyHandle(
        L"PercentIdleTime",
        &PercentIdleTimeType,
        &PercentIdleTimeHandle);
    COMHelper::TestOk(hr);

    // de même pour les autres propriétés

    for (unsigned int i = 0; i < nb_objects; i++)
    {
        UINT64 value;
        VARIANT variant_value;
```

```

cout << "Objet N: " << i+1 << endl;

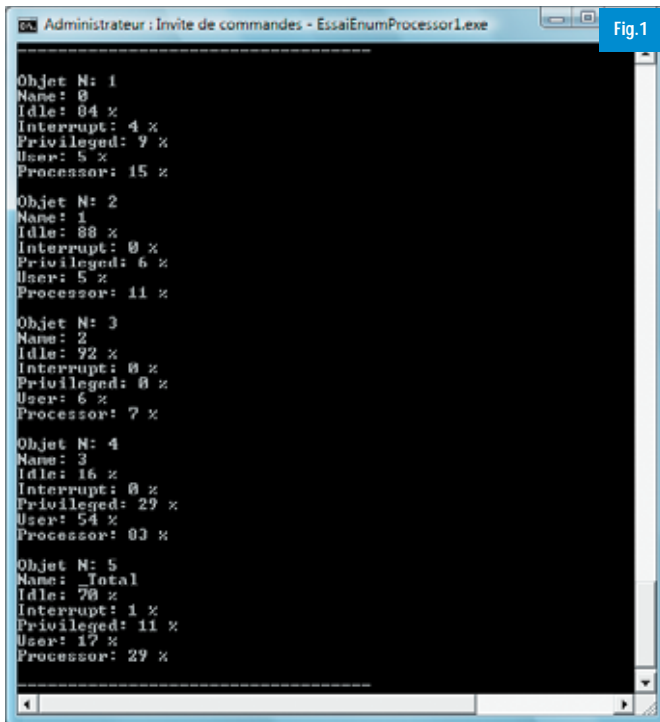
hr = objects[i]->Get(L"Name", 0, &variant_value, 0, 0);
COMHelper::TestOk(hr);
wcout << "Name: " << variant_value.bstrVal << endl;
VariantClear(&variant_value);

hr = objects[i]->ReadQWORD(
    PercentIdleTimeHandle,
    &value);
COMHelper::TestOk(hr);
cout << "Idle: " << value << " %" << endl;

// De même pour les autres propriétés.
}
}

```

Surprise! A l'exécution nous n'obtenons pas cette fois un coeur, ni 4 coeurs, mais 5 :) Que se passe-t-il ? Rappelons nous que nous travaillons avec une classe qui s'occupe de processeurs physiques et/ou logiques. Et ici nous avons bien un processeur physique contenant 4 coeurs (ou processeurs logiques), ce qui donne 5 instances de la classe. Les 4 premières ont comme nom le rang du coeur, la cinquième a comme nom "\_Total" et n'a de total ... que le nom :) Les valeurs qu'elle contient sont en fait la moyenne des valeurs contenues par les quatre autres instances. [Fig.1]



```

Objet N: 1
Name: 0
Idle: 84 %
Interrupt: 4 %
Privileged: 9 %
User: 5 %
Processor: 15 %

Objet N: 2
Name: 1
Idle: 88 %
Interrupt: 0 %
Privileged: 6 %
User: 5 %
Processor: 11 %

Objet N: 3
Name: 2
Idle: 92 %
Interrupt: 0 %
Privileged: 0 %
User: 6 %
Processor: 7 %

Objet N: 4
Name: 3
Idle: 16 %
Interrupt: 0 %
Privileged: 29 %
User: 54 %
Processor: 83 %

Objet N: 5
Name: _Total
Idle: 70 %
Interrupt: 1 %
Privileged: 11 %
User: 17 %
Processor: 29 %

```

Charge de chacun des 4 coeurs, et moyenne des résultats.

## 5 RETOUR À C#

Le même code en C# est plus court :

```

using System;
using System.Management;
using System.Threading;

```

```

namespace CSharpEnumCores
{
    class Program
    {

        private void GetObjects(ManagementClass all_cores)
        {
            String s;
            UInt64 v;
            ManagementObjectCollection cores = all_cores.GetInstances();
            foreach (ManagementObject core in cores)
            {
                s = (String)core.GetPropertyValue("Name");
                Console.WriteLine("Name: {0}", s);
                v = (UInt64)core.GetPropertyValue("PercentIdleTime");
                Console.WriteLine("Idle: {0}", v);
                v = (UInt64)core.GetPropertyValue("PercentInterruptTime");
                Console.WriteLine("PercentInterruptTime: {0}", v);
                v = (UInt64)core.GetPropertyValue("PercentPrivilegedTime");
                Console.WriteLine("Privileged: {0}", v);
                v = (UInt64)core.GetPropertyValue("PercentUserTime");
                Console.WriteLine("User: {0}", v);
                v = (UInt64)core.GetPropertyValue("PercentProcessorTime");
                Console.WriteLine("Processor: {0}", v);
                Console.WriteLine("");
            }
        }

        static void Main(string[] args)<<
        {
            Program p = new Program();
            ManagementClass all_cores =
                new ManagementClass(
                    "Win32_PerfFormattedData_PerfOS_Processor");

            for (int i = 0; i < 20; i++)
            {
                Console.WriteLine("-----\n");
                p.GetObjects(all_cores);
                Thread.Sleep(1000);
            }
        }
    }
}

```

## 6 ETRE AVERTI DE MANIÈRE ASYNCHRONE D'UN CHANGEMENT DE CONDITION

Si les choses vont si bien avec C#, pourquoi ne pas l'utiliser tout le temps ? D'abord parce que l'on a intérêt à ce que le code soit le plus rapide possible lorsque l'on travaille en temps réel. Ensuite, il ne faut pas croire que manipuler la WMI avec C# soit toujours simple. On tombe souvent sur des finesses. Voici un dernier exemple pour illustrer ce fait. Nous voulons être avertis au moment où la charge d'un coeur quelconque dépasse 80%. Nous émettons pour cela une requête WQL. WQL ressemble beaucoup à SQL et permet de voir l'OS comme une base de données. Le lecteur qui souhaite approfondir



# DEVELOPPEZ VOTRE SAVOIR-FAIRE



## 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 !

### Choisissez votre formule

- Abonnement 1 an au magazine : 45 € (au lieu de 65,45 € tarif au numéro) *Tarif France métropolitaine*
- Abonnement Intégral : 1 an au magazine + archives sur Internet et PDF : 57 € *Tarif France métropolitaine*
- Abonnement PDF / 1 an : 30 € - *Tarif unique*  
Inscription et paiement **exclusivement en ligne**  
[www.programmez.com](http://www.programmez.com)
- Abonnement Etudiant : 1 an au magazine : 39 € (au lieu de 65,45 € tarif au numéro) *Offre France métropolitaine*

11 numéros par an : 45 €\*

soit **3 Numéros GRATUITS**

\*Tarifs France métropolitaine

## + Abonnement INTÉGRAL

**ACCÈS ILLIMITÉ aux ARCHIVES du MAGAZINE pour 1€ par mois !** *Prix de lancement*

Cette option est réservée aux abonnés pour 1 an au magazine, quel que soit le type d'abonnement (Éco, Numérique, Etudiant). Le prix de leur abonnement normal est majoré de 12 € (prix de lancement, identique

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

**OUI, je m'abonne**

*Vous pouvez aussi vous abonner en ligne et trouver tous les tarifs [www.programmez.com](http://www.programmez.com)*

### PROGRAMMEZ

- ☐ Abonnement 1 an au magazine : 45 € (au lieu de 65,45 € tarif au numéro) *Tarif France métropolitaine*
- ☐ Abonnement Intégral : 1 an au magazine + archives sur Internet et PDF : 57 € *Tarif France métropolitaine*
- ☐ Abonnement Etudiant : 1 an au magazine : 39 € (au lieu de 65,45 € tarif au numéro) *Offre France métropolitaine*

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

Nom : ..... Prénom : .....

Adresse : .....

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

Tél : ..... E-mail : .....

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

A remplir et retourner sous enveloppe affranchie à :

Programmez ! - Service Abonnements - 22 rue René Boulanger - 75472 Paris Cedex 10.

[abonnements.programmez@groupe-gli.com](mailto:abonnements.programmez@groupe-gli.com)

**Offre limitée,**  
valable jusqu'au  
30 septembre 2008

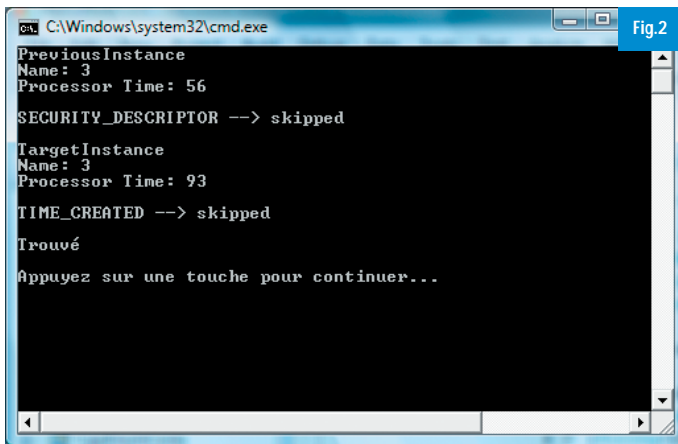
Le renvoi du présent bulletin implique pour le souscripteur l'acceptation pleine et entière de toutes les conditions de vente de cette offre.

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.

**PRO**grammez ! Le magazine du développement



Nous sommes avertis de la montée en charge d'un coeur.

dir cette question est invité à se reporter à Programmez! 79. Quand la condition de la requête est remplie, l'API invoque un gestionnaire d'événements : [Fig.2]

```
using System;
using System.Management;
using System.Threading;

namespace CSharpWatchLoad
{
    class Program
    {
        private bool fini = false;
        private WqlEventQuery query;
        private ManagementEventWatcher watcher;

        Program()
        {
            query = new WqlEventQuery(
                "SELECT * FROM __InstanceOperationEvent " +
                "WITHIN 1 WHERE TargetInstance ISA " +
                "'Win32_PerfFormattedData_PerfOS_Processor' " +
                "AND TargetInstance.PercentProcessorTime > 80 " +
                "AND NOT TargetInstance.Name='_Total'");
            watcher = new ManagementEventWatcher(query);
            watcher.EventArrived += new
                EventArrivedEventHandler(this.ConditionRemplie);
        }

        private void ConditionRemplie(Object sender,
            EventArrivedEventArgs e)
        {
            String s;
            UInt64 v;
            ManagementBaseObject result;
            result = (ManagementBaseObject)e.NewEvent;
            PropertyDataCollection wbem_coll = result.Properties;

            foreach (PropertyData wbem in wbem_coll)

```

```
{
    if (wbem.Name.Equals("PreviousInstance")
        || wbem.Name.Equals("TargetInstance"))
    {
        Console.WriteLine(wbem.Name);
        ManagementBaseObject the_core =
            (ManagementBaseObject)wbem.Value;
        s = (String)the_core.GetPropertyValue("Name");
        Console.WriteLine("Name: {0}", s);
        v = (UInt64)the_core.GetPropertyValue(
            "PercentProcessorTime");
        Console.WriteLine("Processor Time: {0}", v);
    }
    else
        Console.WriteLine("\n{0} --> skipped\n", wbem.Name);
}
Console.WriteLine("Trouvé\n");
watcher.Stop();
fini = true;
}

private void Go()
{
    watcher.Start();
    while (!fini)
    {
        Thread.Sleep(2000);
    }
}

static void Main(string[] args)
{
    Program p = new Program();
    p.Go();
}
}
```

Ici on s'attendrait légitimement à ne voir apparaître que des instances de la classe *Win32PerfFormattedData\_PerfOS\_Processor* dans la collection reçue dans le gestionnaire d'événement. Il n'en est rien. Alors que rien ne le laisse prévoir dans la documentation, deux instances encapsulant respectivement un descripteur de sécurité et un timbre à date se sont glissées dans la collection. Il convient de filtrer ces instances afin d'éviter un plantage à l'exécution, ce qui est effectué dans le test *if* de la boucle *foreach*.

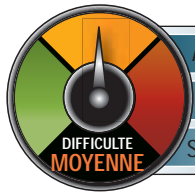
Nous avons ici surveillé la charge d'un coeur, mais rien n'empêche si on le souhaite, d'être alerté d'une élévation de la température de la carte mère ou de n'importe quelle autre condition. Ceci au prix d'une modification mineure du code.

C'est tout l'intérêt de l'API WMI. Le code écrit lors d'un premier abord difficile peut ensuite être facilement repris dans moult situations.

■ Frédéric Mazué  
fmazue@programmez.com

# SQLAlchemy : faites du mapping objet/relationnel en Python

Par définition, un programmeur **Python** aime le code clair et concis... et surtout vite écrit. Pour cette raison, lorsqu'il s'agit de travailler avec une base de données, il évitera volontiers de constituer en dur les requêtes SQL pour préférer un outil de mapping objet/relationnel. **SQLAlchemy est un de ces outils.**



APPLICATION : DONNÉES

LANGAGE : PYTHON

SOURCE : OUI

SQL (Standard Query Language) est le langage permettant de travailler avec les systèmes de gestion de bases de données rela-

tionnelles. Est-ce vraiment un langage ? La question est souvent discutée. S'il est bien question, avec SQL, de former des phrases destinées à être compilées pour exécuter des traitements, certains préfèrent parler de pseudo-langage tant SQL est différent des autres langages informatiques et se marie mal avec eux en général et avec les langages orientés objet en particulier. Quand il s'agit de sauvegarder les données encapsulées par un objet dans une base de données, il est nécessaire de constituer manuellement la requête SQL, ce qui représente un travail fastidieux. C'est ici que l'outil de mapping objet relationnel intervient. Son rôle est de mettre en correspondance les objets avec les lignes d'une table de données et les données membres des objets avec les colonnes de la table. Et bien sûr, c'est l'outil de mapping qui se charge de constituer les requêtes SQL. Ceci ne peut que séduire les programmeurs Python, en principe paresseux comme des couleuvres ;) Il existe deux outils de mapping pour Python, tous deux intéressants : SQLAlchemy et SQLObject. Nous nous familiarisons aujourd'hui avec SQLAlchemy.

## 1 INSTALLATION ET PRÉ-REQUIS

Pour utiliser SQLAlchemy, votre système doit disposer d'un Python version 2.3 strict minimum. Il est bien évident qu'un SGDBR doit être opérationnel. Nous travaillerons sur une base de données baptisée 'Test'. Nos exemples ne créent pas cette base, et vous devez donc vous en charger. En plus de cela, vous devez avoir une implémentation de la DB-API pour ce SGDBR. Programmez! vous a présenté la DB-API de Python dans son numéro 110. A cette occasion nous avons utilisé PostgreSQL, SGDBR favori de votre serveur, et pycopg2 en ce qui concerne l'implémentation de la DB-API. Nous nous basons sur les mêmes outils aujourd'hui, ce qui revient à dire que nous partons du principe que pycopg est présent. Toutefois, notre code d'exemple doit pouvoir être utilisé directement avec le SGDBR de votre choix, pourvu qu'une DB-API pour ce dernier soit également présente. Le script d'installation de SQLAlchemy requiert l'installation préalable d'un ensemble d'autres outils: les *setuptools*. Ceux-ci constituent une suite d'utilitaires de packages Python. Ils se téléchargent à <http://pypi.python.org/pypi/setuptools>. L'installation est une simple formalité. Ceci fait on téléchargera SQLAlchemy à <http://www.sqlalchemy.org/>, puis on décompressera l'archive dans un

répertoire quelconque. Nous avons maintenant deux possibilités pour installer SQLAlchemy. La première, classique, consiste à ouvrir une console, à se positionner dans le répertoire susmentionné et à saisir la commande :

```
python setup.py install
```

La seconde possibilité consiste à passer par l'utilitaire Easy Install qui fait partie des *setuptools*. Après installation de ces derniers, le Easy Install se situe dans le sous-répertoire *Scripts* de votre distribution Python. C'est éventuellement une bonne idée de faire pointer ce répertoire par le PATH de votre système. Ensuite on ouvrira une console, on se positionnera dans le répertoire de désarchivage de SQLAlchemy, puis on saisira la commande :

```
easy_install SQLAlchemy
```

Nous attirons l'attention du lecteur sur le fait que, même s'il choisit la première possibilité d'installation, la présence des *setuptools* est absolument requise. Dans tous les cas on vérifiera le bon fonctionnement de SQLAlchemy en lançant un interpréteur interactif Python et en faisant un petit essai :

```
>>> import sqlalchemy
>>> print sqlalchemy.__version__
0.4.3
```

## 2 UN EXEMPLE SIMPLE

Pour nous familiariser avec SQLAlchemy nous allons suivre une démarche parallèle à celle proposée par le tutorial officiel. Cependant votre serveur regrette que celui-ci n'utilise que l'interpréteur interactif. Si l'on considère qu'un fichier de code Python est ipso facto un module, nous pouvons tirer parti de cela pour une utilisation confortable de SQLAlchemy. Pour notre exemple, il s'agit de travailler avec des membres de l'équipe Programmez! Dans la base de données chaque membre est une ligne dans une table, chaque donnée relative au membre, ici nom prénom fonction et salaire, est une colonne dans la table. Côté Python chaque membre sera une instance de classe, chaque donnée relative au membre de l'équipe étant une propriété de l'instance. Voici le code qui gère tout cela :

**ProgrammezMapper.py**

```
# -*- coding: latin1 -*-

import sqlalchemy
import sqlalchemy.orm as sqlorm
```



```
# Définition d'une classe à mapper
class Membre(object):
    def __init__(self, nom, prenom, fonction, salaire):
        self.nom = nom
        self.prenom = prenom
        self.fonction = fonction
        self.salaire = salaire
    def __repr__(self):
        return "<Membre('%s','%s', '%s', %s)>" \
            %(self.prenom, self.nom, self.fonction, self.salaire)

# Instanciation du moteur
engine = sqlalchemy.create_engine("postgres://fred:xxxx@localhost/Test")
# Création de la table
metadata = sqlalchemy.MetaData()

equipe_table = sqlalchemy.Table('equipe', metadata,
    sqlalchemy.Column('id', sqlalchemy.Integer, primary_key=True),
    sqlalchemy.Column('nom', sqlalchemy.String(40)),
    sqlalchemy.Column('prenom', sqlalchemy.String(40)),
    sqlalchemy.Column('fonction', sqlalchemy.String(100)),
    sqlalchemy.Column('salaire', sqlalchemy.Numeric()) )

metadata.create_all(engine)
sqlorm.mapper(Membre, equipe_table)
# Session est un objet prêt à être utilisé
# pour travailler avec la table
Session = sqlorm.sessionmaker(bind=engine,
    autoflush=True, transactional=True)
```

Outre la déclaration de la classe et de la table, nous trouvons dans ce code l'instanciation du moteur de SQLAlchemy. Celle-ci se fait sur la base d'une URI. Le lecteur est invité à s'informer dans la documentation des particularités relatives au SGDBR qu'il utilise le cas échéant. A la fin du code, l'appel à la fonction mapper "branche" la classe avec la table, puis une classe session est définie. Cet objet sera la charnière de tout travail ultérieur sur les données/classes. Lancer notre script provoque la création de la table dans la base de données. Cette table est bien entendu vide. Lancer le script lorsque la table existe déjà est sans effet sur celle-ci. Adapter ce script à vos besoins particuliers est très simple. Prenez toutefois le temps de lire la documentation afin de bien connaître la correspondance des types Python avec les types SQL.

## 3 INSÉRER DES DONNÉES

Ce travail est immédiat, et SQL qui était quand même présent en filigrane à l'étape précédente se fait maintenant oublier:

### InsertProgrammez.py

```
# -*- coding: latin1 -*-

import ProgrammezMapper as pm

session = pm.Session()
```

```
session.save(pm.Membre("Kaminsky", "Jean", "Big Boss Man", 100000))
session.save(pm.Membre("Tonic", "François", "Rédac'Chef", 1000))
session.save(pm.Membre("Mazué", "Frédéric", "grouillot", 10))
session.commit()
```

Nous utilisons notre module ProgrammezMapper.py ainsi qu'une instance de la classe Session qu'il définit. Par le biais de cette instance nous sauvegardons des instances de notre classe Membre. Ces opérations ont pour résultat l'insertion de lignes de données dans la table SQL. Mais seulement après que la méthode commit ait été invoquée. Cette méthode finalise une transaction SQL.

## 4 INTERROGER LA BASE DE DONNÉES

Toujours en nous appuyant sur notre module ProgrammezMapper.py nous pouvons écrire un script qui interroge notre base de données dans tous les sens imaginables :

### QueryProgrammez.py

```
# -*- coding: latin1 -*-

import ProgrammezMapper as pm

def print_resultat(resultat):
    if(resultat == []):
        print "pas de resultat"
        return
    for m in resultat:
        print m.id, m.prenom, m.nom, m.fonction, m.salaire

session = pm.Session()

print session.query(pm.Membre)

resultat = session.query(pm.Membre).all()

for membre in resultat:
    print membre

print
print_resultat(resultat)
print
requete = session.query(pm.Membre).order_by(pm.Membre.prenom)
print requete
resultat = requete.all()
print_resultat(resultat)

print
requete = requete.filter(pm.Membre.salaire > 50)
print requete
print_resultat(requete.all())

print
requete = requete.filter(pm.Membre.nom.like('K%'))
print requete
print_resultat(requete.all())
```

```

requete.all()[0].salaire = 200000
session.commit()

# ou bien
#print
#import sqlalchemy
#requete = requete.filter(sqlalchemy.and_(
#    pm.Membre.salaire > 50, pm.Membre.nom.like('K%'))
#print requete
#print requete.all()
#print_resultat(requete.all())

```

Chaque invocation à `session.query` aboutit à la création d'un objet qui encapsule la requête SQL sous-jacente. A tout moment cet objet peut se rendre, ce qui nous permet de contrôler le travail. Ainsi :

```

requete = session.query(pm.Membre).order_by(pm.Membre.prenom)
print requete

```

affiche la requête SQL :

```

SELECT equipe.id AS equipe_id, equipe.nom AS equipe_nom,
equipe.prenom AS equipe_prenom, equipe.fonction AS equipe_
fonction, equipe.salaire AS equipe_salaire FROM equipe ORDER
BY equipe.prenom

```

La requête courante est susceptible d'être modifiée à la volée. Ainsi lorsque nous trouvons

```
requete.filter( #etc
```

dans notre script, la requête SQL vue précédemment se trouve enrichie d'une clause WHERE. Deux invocations successives de `filter` équivalent à une clause WHERE travaillant avec l'opérateur logique AND. On peut si on le souhaite travailler directement avec des opérateurs logiques. C'est ce que fait le code placé en commentaire dans notre exemple ci-dessus. Faisons encore deux remarques: tous les éléments constitutifs d'une requête SQL ne sont pas activés par un objet requête, mais parfois par l'objet encapsulant les données. Ainsi en va-t-il pour l'équivalent de l'opérateur LIKE comme on le voit dans notre exemple. Enfin, il est possible de modifier une ligne de table (équivalent du SQL UPDATE) simplement en modifiant une propriété d'instance. Notre exemple le fait pour un salaire. La validation effective dans la table ne venant qu'après un appel à `commit` bien entendu.

## 5 ETABLIR UNE RELATION UN À PLUSIEURS

Il est très rare de travailler avec une seule table. Ainsi les membres de l'équipe de Programmez! écrivent des articles. Les références à ces articles (par exemple le titre et un extrait du texte) seront stockées dans une table, les enregistrements se référant alors au membre de l'équipe par le moyen d'une clé étrangère. Cela s'obtient ainsi, tout en continuant de nous appuyer sur `ProgrammezMapper.py` :

### OneToMany.py

```

# -*- coding: latin1 -*-

import sqlalchemy
import sqlalchemy.orm as sqlorm
import ProgrammezMapper as pm

# classe à mapper dans la relation un à plusieurs
class Article(object):
    def __init__(self, titre, extrait):
        self.titre = titre
        self.extrait = extrait

    def __repr__(self):
        return "Titre: %s\nExtrait: %s\n" % (self.titre, self.extrait)

# la table correspondante
articles_table = sqlalchemy.Table('articles', pm.metadata,
    sqlalchemy.Column('id', sqlalchemy.Integer, primary_key=True),
    sqlalchemy.Column('titre', sqlalchemy.String(100), nullable=False),
    sqlalchemy.Column('extrait', sqlalchemy.Text(), nullable=False),
    sqlalchemy.Column('user_id', sqlalchemy.Integer,
        sqlalchemy.ForeignKey('equipe.id')))

pm.metadata.create_all(pm.engine)

sqlorm.clear_mappers()

sqlorm.mapper(pm.Membre, pm.equipe_table,
    properties={
        'articles':sqlorm.relation(Article, backref='membre_equipe') })

sqlorm.mapper(Article, articles_table)

```

Nous pouvons maintenant insérer quelques données dans notre table. Ensuite pouvons très facilement obtenir tous les articles écrits par un ou plusieurs membres de l'équipe, ou obtenir le nom de l'auteur de chacun des articles répertoriés. Nous renvoyons le lecteur aux exemples `InsertOneToMany.py` et `TestOneToMany.py` sur le CD-Rom ou le site pour la démonstration de ces deux opérations, respectivement. SQLAlchemy présente une formidable puissance, alliée à une non moins formidable souplesse. Toutefois le pythoneur paresseux peut le trouver un petit peu verbeux par rapport à ce qu'il a coutume d'obtenir avec Python. Dans ce cas, il peut être intéressant de se tourner vers Elixir (<http://elixir.ematia.de/trac/wiki>) un outil de mapping de plus haut niveau encore, construit au dessus de SQLAlchemy.

■ Frédéric Mazué  
fmazue@programmez.com

GAGNEZ 300 licences  
Microsoft SQL Server 2008 Edition Standard  
www.programmez.com

## Facebook & OpenSocial



Difficulté : \*\*  
Editeur : Micro  
Application  
Auteur : Loïc Bar  
Prix : 25 €

Sujet particulièrement à la mode, l'OpenSocial n'avait pas encore d'ouvrage de référence, pouvant aider le webmaster, le développeur à s'y retrouver. C'est désormais chose faite grâce à Loïc Bar. Nous sommes bien dans le développement car l'auteur y décrit comment fonctionne FaceBook : son site, les services, et surtout les API et comment on les attaque et les déploie. On découvre ainsi la vraie richesse de l'environnement FaceBook : exploitation des données avec MySQL, intégration avec Google Gadget, Ria avec Flash et Silverlight, etc. Clair et précis ce livre ouvre une nouvelle porte sur le web.

## Management d'un projet système d'information 6e édition

Difficulté : \*\* - Editeur : Dunod  
Auteur : Chantel Morley - Prix : 12 €

Cet ouvrage s'adresse aux responsables de systèmes d'information et aux chefs de projets, ainsi qu'aux étudiants en informatique ou système d'information et aux élèves ingénieurs. Quelle est la meilleure façon de conduire un projet système d'information ? Ce livre répond à cette interrogation en analysant les outils et les méthodes de gestion du domaine à partir de nombreux points clés.

## Modèles et métamodèles

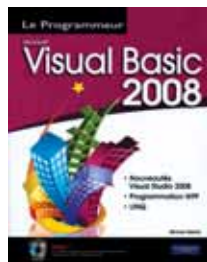


Difficulté : \*\*\*  
Editeur : Presses polytechniques et universitaires romandes  
Auteur : Guy Caplat  
Prix : N.C.

Aujourd'hui, on parle quotidiennement de modèles, de métamodèles. Mais qu'est-ce au juste ? Comment cela fonctionne ? Quels concepts induisent-ils ? Le modèle existe depuis que l'Homme existe. On démarre la lecture par la notion de modèle. L'auteur fixe ou refixe ce qu'est un modèle, à quoi il sert, comment il se présente, les finesses de compréhension. Puis on commence avec la partie

de la représentation. Il ne faut pas oublier qu'un modèle représente quelque chose. Le chapitre suivant aborde le métamodèle. Un chapitre entier est consacré au modèle et au sujet d'étude. Pas toujours évident à suivre car il faut parfois relire plusieurs fois un passage pour être certain de comprendre, mais c'est là aussi la qualité de ce livre. Il remet en place bien des idées reçues sur nos modèles. Une référence pour tout développeur et architecte !

## Visual Basic 2008



Difficulté : \*\*\*  
Editeur : Pearson  
Auteur : Michel Martin  
Prix : 35 €

Vous avez décidé de passer à Visual Studio 2008. Voilà une bonne idée. Mais comment faire pour que cela ne devienne pas un calvaire ? Michel Martin se propose de vous prendre par la main, ou presque. On débute par un petit rappel sur la plateforme .Net, de Visual Studio, les nouveautés de l'éditeur 2008. Puis on rentre dans le vif du code : les menus, les sons, la gestion des fichiers, l'utilisation de LINQ, de WPF, le développement web, la manipulation de Vista... Bref, de quoi démarrer avec Visual Studio 2008. Inclus un CD-Rom.



## WPF

Difficulté : \*\*\*  
Editeur : Eni éditions  
Auteur : Olivier Dewit  
Prix : 39 €

Etes-vous certain de connaître WPF ? Cet ouvrage, outre l'introduction au Framework, aborde de nombreux éléments : le modèle applicatif, la 2D, la 3D, l'interface graphique, la gestion des données, les mécanismes multimédia, la manipulation des textes et documents. La partie XBAP n'est pas oubliée. Pour le développeur voulant se lancer dans WPF, il s'agit aussi de bien comprendre les principes de codage, les conventions comme la séparation code - interface, XAML, etc. Même le format XPS est présent. Un chapitre est consacré au développement de bibliothèques de contrôles personnalisés. Les techniques d'internationalisation sont présentées.



## ADO.Net 3.5 et LINQ

Difficulté : \*\*\*  
Editeur : Eni éditions  
Auteur : Brice-Arnaud Guérin - Prix : 39 €

Ce livre sur ADO.NET et LINQ s'adresse à tout développeur devant maîtriser les mécanismes d'accès aux données de .NET pour construire des applications distribuées. L'ouvrage met l'accent sur les bonnes pratiques de la programmation basée sur les langages C#, VB.NET, SQL et XML. Le lecteur découvre à partir d'exemples spécifiques pour SQL Server, Oracle et Access l'accès en mode connecté, puis le DataSet, les adaptateurs de données et la gestion des exceptions, éléments structurant l'accès en mode déconnecté.

## Conduite de projet web 4e édition

Difficulté : \*\*\*  
Editeur : Eyrolles  
Auteur : Stéphane Bordage  
Prix : 35 €

Comment gérer avec efficacité un projet Web ? Comment préparer au mieux son déroulement et garantir la réussite de chaque phase ? Nourri par la grande expérience de l'auteur, cet ouvrage vous fournira les moyens d'y parvenir, en proposant des solutions concrètes et opérationnelles pour chaque étape. Émaillé de nombreuses check-lists et grilles d'analyse. Cette 4e édition s'enrichit de documents-types, et d'un outil de création de business plan.



## FXRuby

Difficulté : \*\*\*  
Editeur : O'Reilly  
Auteur : Lyle Johnson  
Prix : 30 €

FXRuby est une des bibliothèques les plus connues du monde Ruby pour faciliter le développement d'interfaces graphiques. Les objectifs de l'ouvrage sont : construire une application GUI, créer et gérer les couches complexes, allier GUI, icônes et images, gérer les menus, fenêtres, etc. Si vous ne connaissez pas encore FXRuby, c'est l'occasion de combler cette lacune et vous serez impressionné par les capacités et les performances de FXRuby ! En Anglais.



# Bonne rentrée à tous ! \*



Une sélection d'objets indispensables pour remplir ta trousse pour la rentrée ...

- Edition Open Source
- Open Source Software Assurance
- Services professionnels
- Formations

**LINAGORA**  
[www.linagora.com](http://www.linagora.com)

\* Même si pour certains c'est plus drôle que pour d'autres ! (Postuler sur : [job.linagora.com](http://job.linagora.com) !)





## Pourquoi peindre avec les doigts ?

### Visualisation Java pour clients riches et Ajax

ILOG JViews 8.1, la dernière version de la suite d'outils graphiques Java d'ILOG, couvre l'ensemble des fonctionnalités de visualisation avancée.

ILOG JViews 8.1 offre :

- Des composants graphiques puissants : diagrammes, courbes, tableaux de bord, cartographie, diagrammes de Gantt
- Des services évolués : agencement automatique de graphes, affichage performant pour des jeux de données volumineux

- Plusieurs techniques de déploiement : clients riches, applications Web interactives Ajax, Eclipse/RCP et portails
- Une expertise prouvée dans les industries les plus exigeantes : Informatique, télécoms, transport, énergie et défense.

Testez un de nos produits Java dès aujourd'hui <http://jviews.ilog.com>



Changing the rules of business™