

PROgrammez!

www.programmez.com

Le magazine du développement

La nouvelle galaxie Microsoft

Windows Azure, Windows 7, Visual Studio 2010, .Net 4.0, Oslo... tout ce qui va changer pour le développeur en 2009



Sam
Ramji,
Microsoft

Le meilleur ami
de l'**Open Source** ?

Dossier

PHP

Choisir ses outils
Optimiser ses architectures web.
OpenDocument et PHP
Mes premiers pas avec Symfony

CRÉEZ VOTRE Wiki

Savoir bien collaborer

EPITA

L'école
des ingénieurs
informaticiens

Java 6

Update 10 :
une version majeure
à adopter d'urgence

Visual Studio 2008 SP1 :

toutes les nouveautés
pourquoi faut-il l'utiliser ?

Web

Développez vos applications
OpenSocial

Java

- Comment surveiller vos applications Java ?
- Maîtriser le serveur BlazeDS

.Net

- Regex : les expressions régulières à votre service
- Manipuler et générer des documents OpenXML

3D

Utiliser OpenGL en Ruby !

SGBD

Marier Ultimate++
et PostgreSQL

M 04319 - 114 - F: 5,95 €



Nouvelle version **14**

1 PC pour 1 Euro de +

du 27
octobre
au
21 décembre
2008

**ACHETEZ WINDEV (OU WINDEV MOBILE
OU WEBDEV) CHEZ PC SOFT ET RECE-
VEZ 1 PC DELL POUR 1 EURO DE PLUS**

Cette offre est valable uniquement sur les références produit WD14EE, WM14EE et WB14EE au tarif «catalogue». Aucune remise ne sera appliquée sur ce tarif.

Attention: cette offre n'est pas valable sur les mises à jour, échanges, achats groupés, offres spéciales et échanges concurrentiels.

Offre valable pour la France métropolitaine uniquement, pour les sociétés, administrations et indépendants.

Cette offre est applicable pour toute commande reçue à PC SOFT entre le 20 Octobre 2008 et le 19 Décembre 2008 inclus.

Tous les détails se trouvent sur le site : **www.pcsoft.fr**
(ou appelez-nous, nous répondrons avec plaisir à vos questions)

A PROPOS DE WINDEV 14

Environnement de développement totalement intégré (IDE, ALM) réputé pour sa puissance et sa facilité d'utilisation, WINDEV est livré complet: éditeur d'analyse (ULM,...), RAD, patterns, lien avec toutes les bases de données (ODBC, OLE DB), Oracle, SQL Server, AS/400, Informix, DB2..., lien natif MySQL, base de données Client/Serveur HyperFileSQL gratuite incluse, Générateur d'états PDF, Codesbarres, Accès natif SAP R/3, Lotus Notes, Gestion de planning, Gestion des Exigences, L5G, SNMP, Bluetooth, TAPI, OPC, FTP, HTTP, Socket, Twain, API, DLL, domotique, liaisons série et USB, débogage à distance, profiler, refactoring, génère des applications JAVA à la demande, multilingue automatique, Gestionnaire de versions, Installateur 1-clic, etc... Les applications créées fonctionnent avec toutes les versions de Windows: 98, 2000, NT, 2003, XP, Vista, sous TSE et Citrix, sur eeePC...

**WINDEV : élu
«Langage le plus
productif du marché»
(Octobre 2008)**

CODE MULTI PLATEFORME
Windows, .Net, Java, PHP, J2EE,
XML, Internet, Ajax, Pocket PC,
SmartPhone, Client riche...

**Un PC DELL Vostro
410 + écran DELL 22
pouces E2209W.
Ou, au choix, un portable
DELL Studio
1537 avec lecteur Blu-Ray**
Tous les détails sur
www.pcsoft.fr

Le Logiciel et le matériel peuvent être acquis séparément; WINDEV au tarif catalogue de 1973,40 Euros TTC; le PC et son écran livrés en France métropolitaine au tarif de 1.888,50 Euros TTC et le Portable livré en France métropolitaine au tarif de 1.274,98 Euros TTC (tarifs au 15/10/2008 modifiables sans préavis).

www.pcsoft.fr

Tél province : 04.67.032.032 Tél Paris : 01.48.01.48.88

Fournisseur Officiel de la
Préparation Olympique



sommair

\\ actus

L'actualité en bref	6
Agenda	10
Visual Studio 2008 et .Net3.5 SP1 pour les développeurs	12

\\ projets

Le Model Driven Testing	15
-------------------------------	----

Travailler en équipe,
les outils du succès

Wiki et les développeurs	20
Les outils de collaboration	22

\\ carrières

Epita, l'école d'ingénieurs informaticiens labellisée	24
Transformer des ingénieurs en consultants métier Finance	26

\\ dossier : PHP par la pratique

Panorama des IDE PHP	29
Monitorer et optimiser vos architectures web	30
Manipuler des fichiers OpenDocument	35
Mon premier projet avec Symfony	37

\\ événement :

Microsoft, une nouvelle galaxie

Microsoft dans les nuages ! (compte rendu de la dernière PDC)	42
Visual Studio 2010 et .NET 4.0 : tour du propriétaire !	47
TechEd 2008 : Oslo, Dublin	51

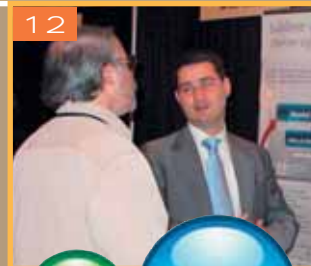
\\ code

Facebook : soyez OpenSocial	52
Ultimate++ et les bases de données	56
Génération de documents en images	60
Health Monitoring : le journal à surveiller	62
Utilisation et exploitation des expressions régulières en .Net	66
Optimisation de la bande passante et messaging temps réel avec BlazeDS	69
Programmer OpenGL avec le langage Ruby	72
Java 6 Update 10 : en attendant Java 7	76
Déconnectez vos applications avec Google Gears	80

\\ temps libre

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

12



22



29



42



52

Adobe
DREAMWEAVER CS4

Développez des applications et des sites web avec le logiciel de création web d'Adobe.
Limitation : 30 jours

.NET

Oslo

Plate-forme de modélisation contenant des outils, un langage pour aider le développeur, l'architecture pour concevoir les applications, un référentiel unique via une base de données.

Silverlight Toolkit

Toolkit fournit un ensemble de contrôles tels que AutoCompleteBox, Chart, TreeView ou encore ViewBox.

ADO.NET Entity framework 2 (32 bits)

Entity Framework est un outil de Mapping Objet Relationnel développé par Microsoft.

SQL Data Services SDK

SSDS est un ensemble de services Web pour le stockage des données et le requêtage de celles-ci.

Microsoft Windows Azure SDK 1.0

Développer sur Windows Azure, plateforme de services disponible sur Internet et hébergée dans les Data Centers de Microsoft.

Azure Tools for Microsoft Visual Studio

Ajoute à Visual Studio des fonctionnalités de développement pour Windows Azure.

Java SDK for .NET Services M1-CTP

Projet open source proposant des bibliothèques Java, exemples et guides pour aider les développeurs Java à construire des applications utilisant les Microsoft .NET Services.

.NET Services for Ruby SDK

SDK open source permettant aux développeurs Ruby d'utiliser les .NET Services.

ODF converter 2.0 (sources)

Permet l'interopérabilité entre les applications basées sur Open Document et OpenXML.

Xwiki

XWiki est un wiki Open-Source de seconde génération écrit en Java (Windows).

Qt creator 0.9

IDE léger et multi plateforme rendant le développement avec le framework Qt plus rapide et aisé (Linux).

Glassfish V3 Prelude

PHP

Wampserver 2.0c

WampServer est une plate-forme de développement web pour Windows.

PHPedit 3.0

IDE de développement PHP (Windows).

Xdebug 2.0.3

Outil de debugage pour PHP.

WinCacheGrind 1.0.0.12

Permet de visualiser les rapports de Xdebug (Windows).

StarUML 5.02

Logiciel Open Source de modelage UML (Linux).

Zend Framework 1.6.2

Bibliothèque de développement pour PHP qui apporte notamment le support de Dojo, une bibliothèque Ajax, et plus de tests unitaires.

CodeIgniter 1.7.0

Framework de développement d'applications Web en PHP.

CakePHP 1.2.0

CakePHP est un framework libre écrit en PHP.



Drupal 6.6

Système de gestion de contenu Web (CMS) développé en PHP.

Magento 1.1.6

Boutique de commerce en ligne Open Source, en PHP.

DES EXPÉRIENCES INTERFACES UTILISATEUR

QUI INSPIRENT



Introducing...



Pour vos applications .NET **actuelles... et futures**

En tant que développeur, votre défi est de concevoir des applications en utilisant les plates-formes les plus matures et robustes disponibles maintenant, Windows Forms ou ASP.NET, tout en planifiant les nouvelles technologies émergentes telles que WPF et Silverlight™. C'est la raison pour laquelle nous alignons nos produits selon vos besoins.

Dans un monde en constant changement, vous pouvez toujours faire confiance aux valeurs durables de NetAdvantage : la puissance et la qualité de ses composants UI ; la documentation efficace ; les tutoriaux instructifs ; l'assistance technique fiable ; et des services de consultation récompensés afin de permettre aux développeurs de livrer à leurs utilisateurs une meilleure expérience dans vos applications d'aujourd'hui et celles de demain.

Pout tout renseignement:
infragistics.com/thefutureisnow

Appelez dès aujourd'hui

N° Vert 0800 667 307

Infragistics®



Errare informaticum est

Qui est apparu le premier, l'ordinateur ou le processeur ? L'ordinateur fait-il le moine ? L'erreur est-elle informatique ou humaine ? Que de questions existentielles fondamentales !

Combien de fois ai-je entendu dans un magasin, au téléphone, " attendez, je n'ai pas la main sur mon logiciel ", " je redémarrer mon PC " ou alors " ah cette option que vous demandez n'est pas disponible pour l'instant "...

N'y a-t-il pas là un paradoxe ? D'un côté on exige de l'informatique la perfection, ou du moins, un fonctionnement impeccable, et dénué d'erreur... De l'autre, on tolère souvent des imperfections, comme un site web qui s'affiche mal, un traitement de texte qui plante ou encore... son téléphone portable avec toutes ces options fort peu ergonomiques.

Dans le domaine du développement, on veut tendre vers le zéro défaut, objectif utopique, reconnaissons-le. On mise - enfin ! - sur les tests, sur la qualité du code.

Dans la plupart des industries, le zéro défaut est la norme : on n'imagine pas une voiture tomber en panne en affichant un message du genre " veuillez patienter... mise à jour du lecteur vidéo... Redémarrage dans 5 min... ". Pourquoi ne pas avoir la même exigence dans les logiciels ? La course aux fonctions, à la nouvelle version, nuit gravement à la qualité finale des logiciels et donc du code. Combien d'heures perdons-nous chaque semaine pour contourner un bug, un problème, trouver une solution parce que l'interface web passe sous Firefox mais pas Safari, etc. ?

On nous montre des PC, des Mac sophistiqués, au design avantageux... Mais le logiciel ? Qu'est-ce que l'on fait pour lui ? Oh, bien sûr, on change le numéro de version, on vous l'affuble des fonctions qui vont bien. On fait trop bling-bling ? On nous colle du " cloud computing " partout alors que personne n'est vraiment d'accord sur sa définition, sur les protocoles et spécifications à utiliser. Bien entendu en geek, en technophile pur et dur, nous sommes excités comme des puces sur un chipset nouvelle génération. Mais avouons-le, l'utilisateur, et le commun des mortels, n'y comprend pas grand chose. Et pour cause. Caractéristiques en franglais, acronymes, concepts philosophico-techniques, injection de code, de dépendances, de classes, de WS-*, de protocoles X ou Y. Même les plus geeks s'y perdent !

Et si finalement, on revenait aux fondamentaux ? Tu codes, donc tu es ! Oh bien sûr, c'est moins sexy que de jouer avec un écran multitouch qui nous lassera dans quelques semaines ... Mais faire un code rigoureux, respectant les bonnes pratiques, n'est-ce pas le premier commandement à appliquer ? Et ça ne se démode pas !

■ FRANÇOIS TONIC

Rédaction : redaction@programmez.com
 Directeur de la Rédaction : Jean Kaminsky
 Rédacteur en Chef : François Tonic
 Ont collaboré : F. Mazué, F. Dewasmes, C. Remy.
 Experts : F. Queudret, G. Mantel, G. Lerouge, R. Lerdorf, S. Haim-Lubczanski, G. Seguy, C. Pierre de Geyer, G. Hubert, G. Renard, L. Bar, A. Petit, X. Millot, G. Lacase, M. Chaize, S. Saurel
 Crédits photo : Php : Johannes Schlüter
 Galaxie Microsoft : World Wide Telescope - F. Tonic
 Maquette : AJE Conseils
 Publicité : Régie publicitaire, K-Now sarl
 Pour la publicité uniquement : Tél. : 01 41 77 16 03
coordination@programmez.com
 Editeur : Go-02 sarl, 6 rue Bezout - 75014 Paris
 Coordination@programmez.com - Dépôt légal :
 à parution - Commission paritaire : 0712K78366
 ISSN : 1627-0908 - Imprimeur : ETC - 76198 Yvetot
 Directeur de la publication : J-C Vaudecrane
 Ce numéro comporte 1 CD Rom

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°115 janvier 2009,
 parution
 30 décembre 2008

• La modélisation de A à Z

Comprendre: UML, MDA, MDD, DSL...
 Les technologies, les outils,
 les formations, et des témoignages !

• Java / Java EE

Quoi de neuf dans l'univers Java ?
 Les meilleures techniques de
 programmation, les dernières librairies !

• Carrière : développeur freelance !

• Open Source

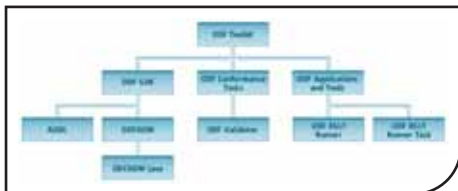
Voyage au cœur du noyau Linux

■ **MX-DEV.Net** propose jusqu'au 31 décembre un concours de développement. Le but est d'être original, utile et ergonomique ! Le site donne en exemple un logiciel de reconnaissance de caractère. Les gagnants se verront offrir un bon d'achat Fnac. A vos claviers. Site : <http://www.mx-dev.net/concours/>

■ **Lyria (groupe W4)** se prépare à sortir la version 4.1 de son outil Leonardi. Cette version apportera trois grosses nouveautés : l'intégration d'Hibernate, la possibilité de générer automatiquement la documentation utilisateur (à partir du modèle de données) et la présence du Document Access Layer (connecteurs GED). Site : <http://www.lyria.com>

■ **COBOL-IT** propose depuis la fin novembre un compilateur COBOL gratuit et open source, COBOL-IT Compiler Suite. Cette annonce est intéressante pour les entreprises utilisant toujours Cobol.

■ **IBM et Sun** travaillent ensemble au projet ODF Toolkit. L'objectif est simple : créer des outils et bibliothèques de dévelop-



pement pour le format ODF (sans rapport avec les suites OpenOffice et consorts). Car avoir un format documentaire normalisé c'est bien, mais pouvoir l'implémenter rapidement et simplement c'est encore mieux car le manque d'outils est rédhibitoire. Projet à suivre ! site : <http://odftoolkit.org/>

■ Succès pour la dernière édition des **Valtech Days** qui se sont déroulés fin octobre. Ce sont plus de 300 participants sur les deux jours. La première journée fut surtout consacrée aux sessions techniques et technologiques, la seconde fut plus orientée retour terrain, question sur des problèmes concrets. De nombreux thèmes furent abordés, dont le cloud computing, les ESB open source, le lean software development. Rendez-vous en 2009 !

Conférence

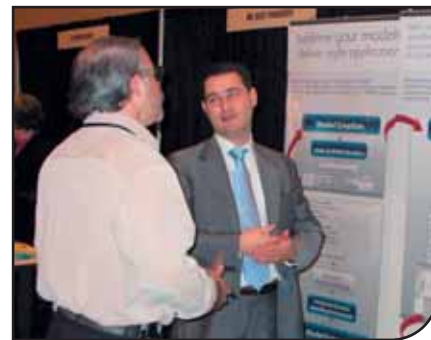
Eclipse World 2008 (Washington)

Eclipse World s'adresse à une communauté ciblée, c'est le repère des aficionados et des experts ! Au programme, 8 workshops et 64 conférences. Occasion de voir l'avancement des différents projets. Mike Milinkovich directeur de la fondation Eclipse, déroule les futurs programmes, dont Eclipse 4.0. A notre surprise la démonstration s'exécute sur un navigateur internet. C'était également l'occasion de voir les premiers effets, s'il en est, des deux annonces fortes de l'année : l'arrivée de Microsoft au sein d'Eclipse et le premier rapprochement opérationnel entre l'EMF et l'OMG.

Au sein de cette diversité des sujets les points forts furent :

- Eclipse 4 : la version web, la gestion des plug-in avec OSGI
- Le développement de plug-in à vocation commerciale
- Les clients riches
- La génération par les modèles

En l'espace de quelques semaines Microsoft a annoncé son intérêt pour UML, sa participation à l'OMG et son attrait pour l'IDE Eclipse. Ces annonces avaient amené IBM à demander à l'OMG de préciser les modalités des votes lors des RFC ; il était donc intéressant de voir si Microsoft serait présent à Washington ou s'il compte l'être à Santa Clara. Or, si le premier n'était pas là, le second est déjà en train de reprendre l'initiative. IBM était bien présent pour expliquer la valeur du statut Ready 4 Rational. En effet Eclipse évolue et se transforme : hier l'IDE Java de réfé-



rence, demain le socle des solutions logicielles d'ALM (application life cycle management). Workshop sur l'écriture de plug-in (" Building Commercial-Quality Plug-Ins for Eclipse "), évolution pour faciliter l'intégration et la gestion des plug-in (OSGI et equinox p2) se produisent au moment où de nombreuses solutions professionnelles se présentent sous forme de workbench reposant sur Eclipse (Flex Builder, IBM Rational, Borland, Blu Age). La convergence se met donc en place, les outils basés sur Eclipse et compatibles avec Rational participeront alors à une offre multi-plate-forme et complète d'ALM. Autre sujet de compétition : les applications riches. Flex et Silverlight (eclipse4sl) sont tous deux intégrés dans Eclipse et sont les sujets de demain.

A noter que pour la première fois le platium sponsor de l'évènement est un éditeur de logiciel Français : Blu Age ... Cet éditeur est également platium sponsor d'EclipseCon à Santa Clara au mois de mars 2009. La France serait elle le leader du MDA ?

Agilité

Serena prépare un outil agile

L'éditeur Serena prépare un outil tout spécialement dédié aux " agilités ", pour les méthodes agiles. Jusqu'à présent, il n'existait pas d'outils spécifiques. Avec *Agile on demand*, Serena veut combler ce vide. Il se présente sous forme SaaS et est accessible en ligne dans son navigateur. Les outils actuels en ALM ou des envi-

ronnements de type MS Project sont inadaptés aux méthodes agiles. L'originalité est de pouvoir gérer aussi bien la méthode agile que l'approche traditionnelle des projets, le référentiel étant commun aux deux. L'objectif est d'offrir un outil léger, facile à déployer, à paramétrer. Ayant en interne des gours de Scrum, Serena a

su tirer profit de cette expérience pour proposer un outil adapté aux agilités qui veulent aller vite. L'éditeur s'appuie en Angleterre et aux USA sur le français **Valtech** qui possède une forte expérience dans les méthodes agiles. En France, la même démarche est en cours. Sortie de l'outil : mi-janvier 2009.

Nouvelle version **14**

501

NOUVEAUTÉS

Réussissez tous vos projets
avec l'outil de développement
le plus productif du marché*.

VERSION
EXPRESS
GRATUITE
Téléchargez-la !

14

Mashup

Lien Google

Lien Salesforce

HyperFileSQL : full
text

DataBinding

Nouveaux
graphiques

Nouvelles tables

Robot de moni-
toring & surveillance

Accès Natif
PostgreSQL

Lien Silverlight 2
et Flex

PHP 5

214 Nouveautés
fonctionnelles

120 Nouvelles
fonctions
WLangage

62 Nouvelles
fonctions Java

32 Nouvelles
fonctions PHP

101 Nouvelles
fonctions LINUX

PLATEFORME INTÉGRÉE
DE DÉVELOPPEMENT
Windows, .Net, RAD Java
Développez 10 fois plus vite

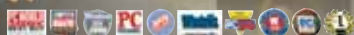
WINDEV®

14



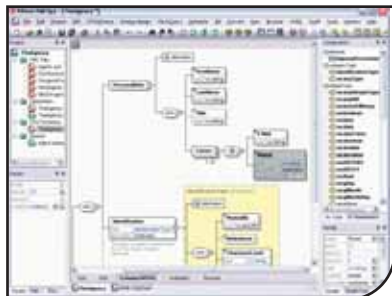
www.pcsoft.fr

Demandez le dossier 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



Fournisseur Officiel de la
Préparation Olympique

■ **L'éditeur d'outils XML, Altova**, vient de dévoiler une suite XML, database et UML, tout intégrée, pour Visual Studio / .Net, tout en proposant le support d'Open XML, de SQL Server et de nombreux outils Microsoft. Altova MissionKit



convient aussi bien aux développeurs XML qu'aux architectes logiciels. Cette suite inclut les outils de l'éditeur : XMLSpy, MapForce, StyleVision, UModel, DatabaseSpy.

■ **Parasoft Embedded** (division de Parasoft) a annoncé une nouvelle version de son outil de tests et de qualité C++, C++ Test. Destiné avant tout aux logiciels embarqués, cette version propose un nouveau Framework de tests unitaires, le support de nouveaux systèmes embarqués et une amélioration des capacités d'analyses statiques du code.

■ **Innooprac**, éditeur allemand, propose une approche originale et efficace d'Eclipse : construire, en ligne, sa propre distribution Eclipse avec le projet open source yoxos. L'environnement gère les dépendances, propose l'ensemble des plug-in, éditeurs, puis il génère la distribution, qui est immédiatement opérationnelle (enfin après la génération). Site : www.yoxos.com

■ **Glassfish Enterprise Server 3.0 Prelude**, le serveur d'applications JEE référence de Sun, arrive sur les serveurs. C'est une version commerciale avec support / assistance. Il s'agit tout de même d'un Glassfish 3 complet, à savoir : architecture basée sur OSGi, environnement de développement revu et corrigé, support des langages dynamiques, API directement embarquées, centre de mise à jour disponible.

Microsoft et l'open source : les meilleurs amis du monde ?



Microsoft a pour certains l'image d'un ogre. Pourtant, l'éditeur a su assouplir ses positions, et participer à la communauté open source, via des partenariats.

La mobilisation des développeurs et les dernières annonces autour de l'OMG, d'Eclipse et des autres langages, marquent incontestablement un tournant important, voire, majeur. On pourrait résumer la première approche de Microsoft par un raisonnement simple : *chercher le business où qu'il soit*. Il y a un peu plus d'un an, lors d'un passage à Paris, **Sam Ramji**, directeur senior de la stratégie plate-forme, et un des experts open source de la société, avait clairement fait comprendre qu'il n'avait

aucun problème à travailler sur des projets open source si cela permettait de garder Windows comme plate-forme serveur et/ou cliente. Un an plus tard, de retour à Paris à la mi-novembre dernier, Sam n'a pas changé de discours, même si aujourd'hui le spectre s'est agrandi avec des accords plus profonds avec Apache, Samba et des perspectives plus qu'excitantes vers Eclipse, l'OMG (autour d'UML). Le support de langages et d'outils non Microsoft sur la plate-forme virtualisation et serveur, annoncé durant les TechEd de

Barcelone, confirme la tendance : ouvrir, supporter, proposer. Bref interopérer avec l'extérieur, meilleur moyen de garder les utilisateurs. Et si cela ne suffit pas : on finance un projet open source pour supporter nos technologies sur des outils non Microsoft.

Objectif Eclipse

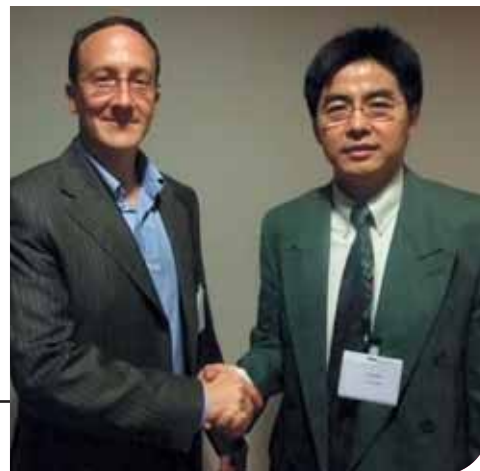
Depuis le printemps dernier, le couple Microsoft - Eclipse n'est plus tout à fait une illusion d'optique. Oh, bien entendu, d'ici à voir Microsoft dans la fondation, un océan passera sous les ponts (et certains membres de la fondation ris-

Aux origines du projet Eclipse4sl

Tout avait commencé au printemps dernier, à EclipseCon 2008 aux USA, où Sam Ramji, responsable chez Microsoft de l'Interopérabilité et de l'Open Source, fait une annonce retentissante. Steve Sfartz, architecte chez Microsoft France raconte : " Il déclare que l'éditeur est prêt à épauler les équipes Eclipse dans le but de supporter la plate-forme Vista en WPF. L'objectif est de mettre en place un support natif de WPF dans SWT, le kit graphique d'Eclipse. " Cette main tendue de Microsoft va se concrétiser sur Silverlight, qui est cross-browser et cross-plateforme. La représentation graphique de l'outil repose sur XAML. Or une société française, Soyatec, a développé eFace, le seul runtime existant qui permet de développer des applications XAML sur Eclipse, et de les exécuter avec Java. A Paris, Steve Sfartz repère ce précurseur. Microsoft Corp lui passe commande des **Eclipse Tools 4 Silverlight**, environnement de développement pour des applications professionnelles Silverlight sous Eclipse. Les parties ne communiquent pas le montant de la tran-

saction, mais expliquent le modèle de collaboration : " c'est un co-sponsoring, Soyatec apporte des briques d'eFace. Il s'agit d'une contribution, sous licence EPL. Le copyright reste acquis à Soyatec " explique Yves Yang, dirigeant de Soyatec. L'entreprise compte une trentaine de personnes, avec une R&D à Paris et l'essentiel des troupes -26 personnes-développent en Chine. " 7 à 8 ingénieurs travaillent sur ce projet ", précise Yves Yang.

■ Jean Kaminsky



quent de ne pas apprécier une telle venue) mais un rapprochement sensible s'observe. Et c'est encore une fois Sam Ramji qui l'a répété durant la grande messe d'EclipseCon ! Un des projets menés en collaboration tournait autour de Cardspace et du projet Higgins pour la sécurité et l'interopérabilité dans la gestion des identités sur le web. L'autre travail qui nécessitera une collaboration étroite concerne l'optimisation de Java sur Windows (Vista et maintenant Seven). Et durant l'EclipseCon, nous avons aussi appris qu'un groupe de développement (provenant de l'équipe SWT) travaillait sur le couplage SWT - WPF et Microsoft annonçait rapidement l'aide de l'Open Source Software Lab... Plus loin, il s'agit d'imaginer le codage natif dans Eclipse d'une application .Net (en C# par exemple). Comme nous l'a précisé Sam, techniquement, c'est possible et avec l'arrivée de Silverlight sur Eclipse, plus rien ne l'interdit ! Et l'éventualité d'avoir un Eclipse pour C# n'a rien d'utopique, bien au contraire !

Les langages

PHP fait partie des autres priorités selon Sam Ramji. On se souvient de l'accord avec Zend pour mieux optimiser PHP sur IIS et Windows Server 2008. Cela passe notamment par le projet ADOdb permettant à PHP de travailler directement avec SQL Server. Dans IIS 7, on dispose de FastCGI pour améliorer la compatibilité et les performances de PHP. Et comme vous le verrez dans notre dossier spécial Microsoft, Windows Azure ne se limite pas aux applications .Net ! Un kit pour Ruby et Java est déjà disponible et la liste n'est pas prête de s'arrêter là ! Pour preuve, le langage M d'Oslo est mis en Open Specification Promise et cela devrait faciliter l'ajout de schémas de modèles et l'extensibilité d'Oslo dans le futur. Enfin, toujours

courant novembre, Sam a confirmé durant une grande conférence Apache, que Microsoft collaborait au projet Apache Qpid qui est une implémentation multi langage du AMQP (protocole de messaging indépendant). Et le second projet, en cours de définition, est le projet Apache Stonehenge. Il s'agit d'un ensemble d'applications types pour les architectures de services basés sur les bonnes pratiques et l'interopérabilité.

Administrer autre chose

Si aujourd'hui, l'éditeur est bien décidé à s'imposer sur les marchés de la virtualisation (serveur, desktop, applicative), Microsoft n'oublie pas de s'ouvrir. Déjà deux importants accords avaient été signés avec Novell et Sun. Aujourd'hui, on parle beaucoup de Cross Platform Extensions dans System Center Operations Manager. Il s'agit de pouvoir voir et gérer des systèmes non Microsoft directement dans System Center. D'autre part, l'éditeur a dévoilé une nouvelle pré-version des connecteurs pour OpenView, Tivoli Entreprise Console.

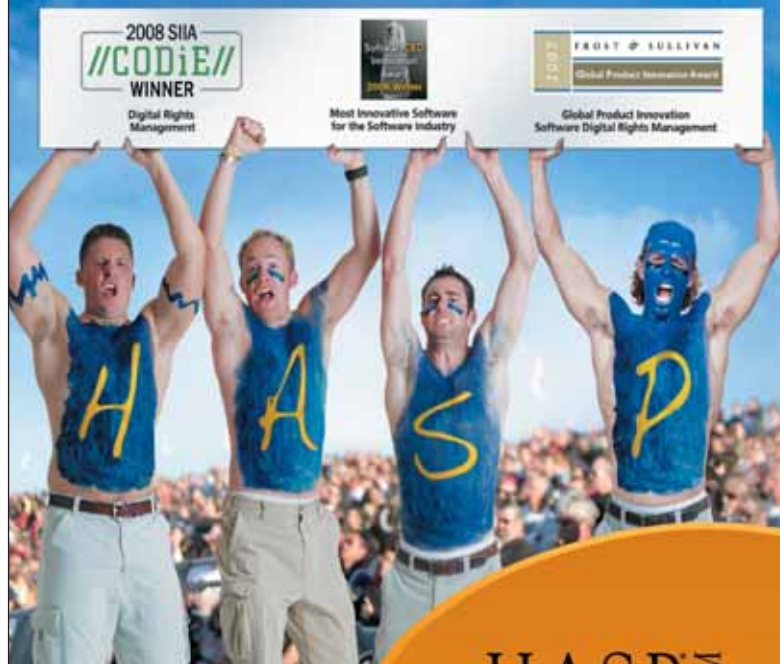
Apache plus que jamais partenaire

Axe stratégique, Microsoft est devenu officiellement un des sponsors de la fondation Apache. Bien entendu, il ne s'agit pas de remplacer son serveur web IIS par de l'Apache mais plutôt de contribuer, de travailler à divers projets : POI, Axis2, Jakarta... POI est un projet qui ne fait pas beaucoup parler de lui mais qui est pourtant important : il s'agit de disposer d'une librairie Java pour manipuler les binaires OpenXML ! Quelle limite dans ce flirt avec l'open source ? Il n'en existe pas de réelle et avec l'émergence des applications et services en ligne, Microsoft devra être toujours "open source compatible"...

■ François Tonic

Ils ont choisi leur camp.

Rejoignez l'équipe gagnante en choisissant la solution **DRM plébiscitée par les professionnels de référence du secteur**



HASP SRM
SOFTWARE RIGHTS MANAGEMENT

Protection logicielle robuste.

Gestion des licences sûre et flexible.

Protégez et valorisez vos logiciels avec

HASP SRM: le choix numéro un des éditeurs de logiciels à travers le monde.*

- Protection logicielle contre le piratage et l'utilisation illégale
- Protection de votre Propriété Intellectuelle contre l'ingénierie inverse et le vol
- Création de nouveaux modèles de vente avec une gestion des licences sûre et flexible
- Protection complètement intégrée et gestion du cycle de vie de vos logiciels

mandez votre kit développeur **HASP SRM GRATUIT** sur:

*Frost & Sullivan N1AF-70, IDC #34452

- FRANCE • +33(0)141377030
- ROYAUME-UNI • AMERIQUE DU NORD • ISRAEL
- BENELUX • ALLEMAGNE • ESPAGNE
- ITALIE • INDE • CHINE
- JAPON • PORTUGAL

Nouveau!
Version 3.50
avec le cryptage des applications JAVA

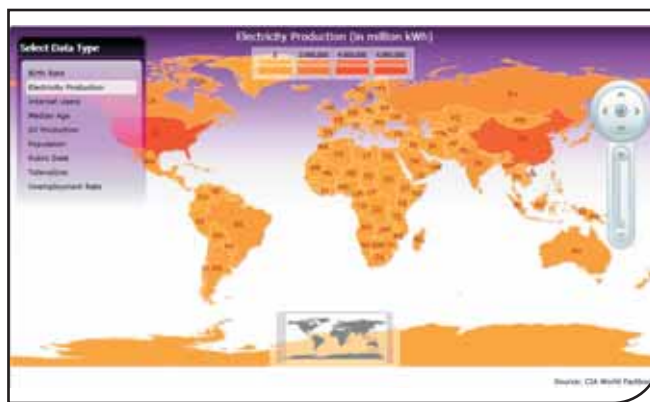
Aladdin
SECURING THE GLOBAL VILLAGE

© Aladdin Knowledge Systems, Ltd. All rights reserved. Aladdin and HASP are registered trademarks. HASP SRM is a trademark of Aladdin Knowledge Systems, Ltd.

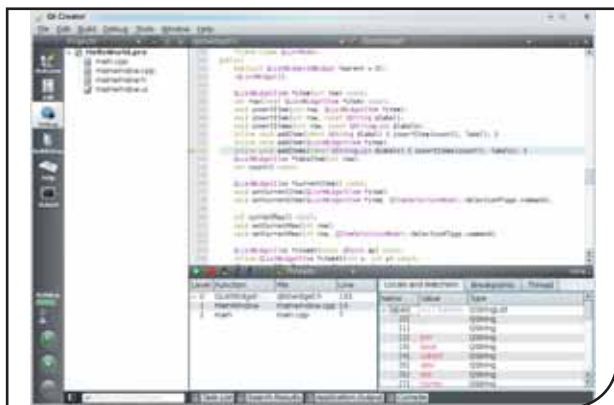
Aladdin.fr/HASP

Composant Infragistics se prépare à Silverlight 2

L'éditeur de composants Java et .Net, Infragistics se prépare à lancer début 2009 une nouvelle gamme de composants pour Silverlight 2.0. Ceux-ci apporteront de puissantes fonctions de visualisation des données à la volée (ou non). Les effets proposés sont assez spectaculaires, avec une fonction zoom très performante. La cartographie est supportée. Ces composants sont disponibles en CTP. Aujourd'hui, la gamme NetAdvantage couvre Windows Form, ASP.Net, WPF. Sur la mobilité, la question se pose mais encore faut-il bien la cerner et pour le moment rester prudent, pareillement sur le multitouch qu'introduit Windows 7, l'adoption risque d'être assez lente. D'autre part l'éditeur nous a confirmé que pour le moment, la priorité reste .net et Silverlight avec une intégration profonde des outils de l'éditeur. Le support des technologies Adobe n'est pas à l'ordre du jour, pareillement pour un



support de Moonlight et d'Eclipse. Outre la partie Data Visualisation, l'éditeur prépare pour le 2e trimestre 2009 des composants Silverlight orientés entreprise / business : NetAdvantage Silverlight avec des composants grilles, arbres et graphs particulièrement puissants et flexibles ! Vivement 2009. Site : <http://www.infragistics.com>



Outil Un nouvel IDE pour Qt !

Sur le CD
du numéro 1

Qt Software (anciennement Trolltech) vient de lancer une pré-version de son nouvel IDE pour développer des interfaces Qt : Qt Creator. Il s'agit d'un IDE léger et multi-plateforme pour concevoir rapidement des interfaces basées sur Qt 4.x. L'outil se focalise sur la productivité

même si on code beaucoup... Il intègre cependant des assistants, une aide contextuelle, un constructeur visuel, un mode debug plus rapide, des fonctions de build, etc. Disponible pour Windows, Linux et MacOS X. Essayez-le dès aujourd'hui ! Disponible sur notre CD-Rom !

■ **Netbeans 6.5** sera disponible depuis plusieurs jours quand vous lirez ces lignes. Cette version améliore un grand nombre de fonctions dans le développement web (PHP, javascript, Ajax, grails, groovy, rails). La partie mobile subit un lifting bienvenu avec de nouveaux composants et assistants. Bref, une version de consolidation de la 6.1 avec des améliorations qui devraient séduire de nombreux développeurs...

■ **Ilog** annonce le support de Visual Studio 2008 et du .net 3.5 très prochainement dans Ilog Rules for .net, système de gestion de règles métiers. Par contre, la nouvelle version est dès maintenant certifiée Windows Server 2008.

RIASPHERE, le portail des interfaces riches

RiaSphere a démarré le 28 octobre. Le portail est lancé à l'initiative de l'éditeur Ilog. Mais Alexandre Videt, Responsable marketing et communication Europe de la gamme des logiciels de visualisation d'Ilog (dont JViews, les logiciels pour

.Net et Elixir) et pilote du projet, insiste sur la dimension communautaire du site. L'objectif est de « créer un lieu d'échange et de rencontre autour de la visualisation et du RIA ». Le portail est clairement international,

uniquement en anglais pour le lancement, mais devrait avoir sa version française début 2009. Les promoteurs du projet espèrent une audience de 20 000 visiteurs mensuels, d'ici juin, dont 20% dans l'hexagone. www.riasphere.org

agenda \

DÉCEMBRE

1er décembre, **Paris On Rails** : la plus grande conférence française sur le langage Ruby et le Framework Rails ! <http://paris.onrails.info/>

1er et 2 décembre, Paris, maison de la chimie. Les conférences, ateliers et événements du **Forum Mondial du Libre** réuniront les meilleurs experts et acteurs du FLOSS. <http://www.openworldforum.org/>

1er décembre, **Nuxeo Developer Day** : dans le cadre du Forum Mondial du

Libre, Nuxeo organise sa première grande conférence développeur ! <http://www.nuxeo.com/en/events/first-nuxeo-developer/>

Les 8 et 9 décembre, Paris : **Forum PHP**, deux jours pour découvrir, parler de PHP, les dernières annonces, les dernières versions. <http://www.afup.org/pages/forum-php2008/>

Jusqu'au 9 décembre, Tour de **France technique WinDev 14** : à l'occasion de la version 14 de WinDev, PCSoft organise son tour de France, tous les détails sur :

<http://www.pcsoft.fr/pcsoft/tdf-com/2008/index.html>

Jusqu'au 17 décembre, tour de **France Cisco**. Pour faire le point sur l'offre Cisco, les nouveaux risques, la sécurité. <http://www.cisco.com/web/FR/minisites/cappme/index.html>

ETRANGER

Du 1er au 4 décembre, **Adobe Max, Milan** : la grand messe Adobe sur Ses technologies et les prochaines versions et projets des labs ! <http://max.adobe.com>

Goût du
contact

Dynamisme

Curiosité

Pédagogie

Esprit de
synthèse

Expertise

Valtech Training recrute...

Formateurs Java / Java EE

Profils

Concepteur développeur
Architecte technique



Formateur UML

Profil

Analyste concepteur



Formateurs .Net

Profils

Concepteur développeur
Architecte logiciel



Visual Studio 2008 et .NET 3.5 SP1 pour les développeurs

Ces service packs auraient pu être baptisés "feature" packs tellement les apports de nouveautés sont nombreux. Présentés par Microsoft comme un jalon significatif, les *service packs 1* de Visual Studio 2008 et .NET Framework 3.5 ont été conçus suivant 3 lignes directrices : améliorer les techniques de développement des applications, simplifier la programmation orientée données et améliorer le déploiement des applications.

Ce SP1 est un jalon dans l'histoire des outils et plates-formes de développement de Microsoft, il est disponible pour .NET 3.5, Visual Studio 2008, les versions Express de Visual Studio et Team Foundation Server 2008. Nous détaillerons principalement dans cet article les améliorations et nouveautés apportées à Visual Studio et .NET.

Le SP1 de Visual Studio 2008

Le SP1 est disponible en téléchargement complet au format ISO (environ 830Mo) ou en installation continue via le Web. Si vous aviez préalablement installé des services packs bêta ou hot fixes, vous pouvez utiliser un utilitaire pour préparer l'installation du SP1. Attendez-vous à quelques heures d'installation avant de pouvoir goûter aux nouveautés annoncées :

- Amélioration du support de SQL Server 2008 : les assistants, projets & concepteurs de données intègrent maintenant SQL 2008.
- Disponibilité d'un concepteur pour ADO.NET Entity Framework.
- ASP.NET : ajout de nouveaux templates et support de l'IntelliSense pour JavaScript. [Fig.1]
- Windows Presentation Framework :
 - Les événements des contrôles sont maintenant accessibles depuis la fenêtre des propriétés.
 - Amélioration du designer WPF "Cider" pour les contrôles Grid, TabControl et Expander, et aussi la simplification du dimensionnement pour les marges.
 - Améliorations du refactoring XAML, et de la recherche des références et définitions dans le XAML. [Fig.2]
- LINQ : le debugger supporte maintenant l'évaluation des expressions.
- Disponibilités de nouveaux contrôles Windows Forms (Vector shape, DataRepeater et Printing) et MFC C++ 2008 (Ribbon...).

Le SP1 de .NET Framework 3.5

Le *service pack* pour .NET n'est pas en reste d'améliorations et de nouveautés :

Pour le développement d'applications .NET :

- Amélioration de 25%-40% de la performance des applications WPF sans changement de code. Des innovations pour WPF

comme les contrôles DataGrid, WebBrowser, Calendar/DatePicker, un ruban "à la Office" et une intégration des API Direct3D.

- Améliorations de WCF tant pour la performance que pour les diverses possibilités d'accéder aux données et services.
- Un nouveau concept de déploiement appelé " .NET Framework Client Profile ", intégré à Visual Studio 2008 SP1 et définissant un sous-ensemble des assemblages .NET à déployer. Ceci permet d'optimiser la taille du redistribuable .NET pour un déploiement client (environ 28 Mo de redistribuable pour installer .NET ; c'est 86% de moins qu'avant). Les types d'applications ciblées (dont les assemblages font partie du sous-ensemble déployé) sont Windows Forms, WPF et WCF. [Fig.3]

Pour le développement d'applications orientées données :

- ADO.NET Entity Framework : une évolution d'ADO.NET permettant d'abstraire les structures de données sous forme d'entités. Cet 'Entity Data Model' permet aux développeurs d'être plus proches de la sémantique conceptuelle de l'application et de réduire le code technique. Les développeurs pourront utiliser 'LINQ to Entities' pour requêter les entités métier ou objets de données fortement typés.
- ADO.NET Data Services : ce framework permet de développer des services REST pour exposer des données à des clients Web tels qu'ASP.NET, Ajax ou Silverlight.

Pour le développement d'applications Web :

- ASP.NET Dynamic Data : un framework qui permet de développer très rapidement des applications Web orientées données. Le modèle de données est découvert à l'exécution et le comportement de l'interface Web est adapté en conséquence.
- ASP.NET Routing Engine : un composant clé du modèle MVC d'ASP.NET ; il permet de définir des mapping ou " routes " entre des URL et les fonctions de vos applications.
- Support de l'historique du navigateur client depuis ASP.NET.
- JavaScript script combining : cette fonctionnalité permet de réduire le nombre de fichiers JavaScript à télécharger par le navigateur (combinaison de plusieurs fichiers .JS en un seul) et donc d'améliorer la vitesse de chargement des applications AJAX par exemple.

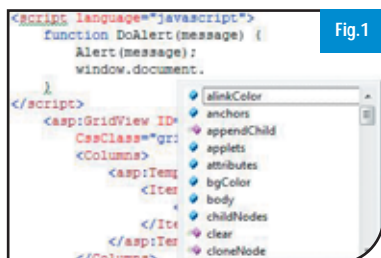


Fig.1



Fig.2

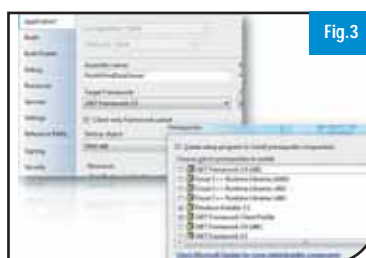


Fig.3

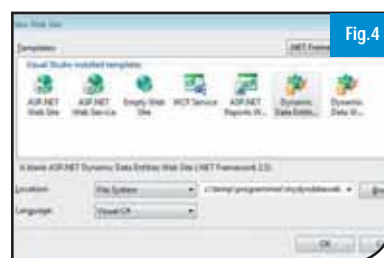


Fig.4

**FarPoint Spread for ASP.NET** à partir de € 700

FarPoint

Composant de feuille de calcul ASP.NET haute performance personnalisable.

- Nouvelles fonctions : extensions AJAX, impression vers PDF, éditeur de modèle de ligne, assistant de démarrage rapide, nouveaux types de cellules, etc.
- Modes liés et non liés (aucun ensemble de données nécessaire), AJAX, import/export Microsoft Excel natif, édition en cellule, redimensionnement client, etc.
- Plus de 300 fonctions de calcul intégrées

**ReSharper** à partir de € 155

JetBrains

Outil de productivité enfichable et intelligent pour Visual Studio 2005/2008.

- Assistance intelligente au codage, signalement des erreurs en cours de travail et correction rapide
- Refonte du code, test des unités, navigation et recherche, édition de scripts NAnt et MS Build, édition ASP.NET, etc.
- Toutes ces fonctions avancées sont disponibles dans Visual Studio
- Analyse et signale les erreurs de code C# (jusqu'à C# 3.0) en cours de frappe

**DXperience Enterprise** à partir de € 984

DevExpress

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

**ComponentArt WebChart for ASP.NET** à partir de € 605

ComponentArt

Fonction avancée de schémas 3D pour ASP.NET AJAX.

- Moteur de rendu haute qualité, pour une image supérieure
- Moteur de grande vitesse, pour un rendu de 5 à 10 fois plus rapide
- Les utilisateurs finaux personnalisent l'aspect des tableaux et en modifient l'ordre
- Zoom et défilement AJAX pour une navigation rapide
- Arborescences de séries, visualisation réaliste des relations entre les données

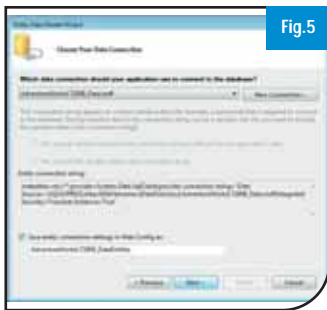


Fig.5

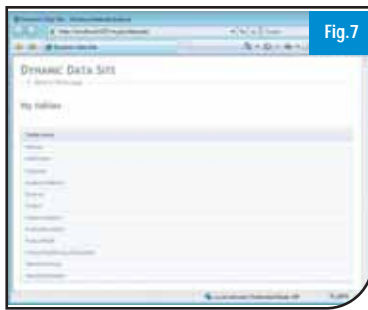


Fig.7

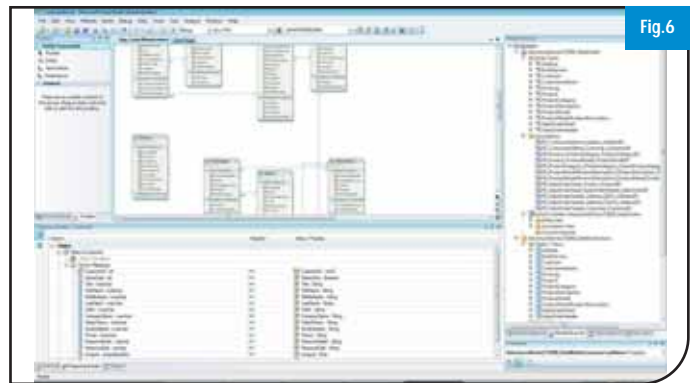


Fig.6

Un peu de pratique avec ASP.NET Dynamic Data et l'Entity Framework

Dans cette partie, nous utiliserons la puissance de Visual Studio 2008 pour créer très rapidement un site Web qui va nous permettre d'afficher et modifier les données d'une base avec quasiment pas de code.

Création du site Web

1. Démarrer Visual Studio 2008 SP1, puis créer un site Web (File > New > Web Site).
2. Sélectionner le template 'Dynamic Data Entities Web Site' puis OK. [Fig.4]. Nous utiliserons ici l'ADO.NET Entity Framework. Une autre possibilité aurait été de créer le site Web basé sur le template 'Dynamic Data Web Site' pour utiliser LINQ to SQL.
3. Nous allons ajouter les données au site Web ; Si vous ne voyez pas de répertoire 'App_Data' dans l'explorateur de solution, faites un clic bouton droit sur le nom de la solution puis 'Add ASP.NET Folder > App_Data'.
4. Cliquez sur le répertoire spécial App_Data avec le bouton droit de la souris puis 'Add Existing Item...'. Sélectionnez le fichier 'AdventureWorksLT2008_Data.mdf' dans le répertoire 'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLEXPRESS\MSSQL\DATA' ou toute autre base que vous souhaiteriez utiliser (Cette méthode copie la base de données dans le répertoire App_Data et n'est utilisée qu'à des fins d'exemple).

Création du modèle de données pour l'Entity Framework

1. Si le site Web ne contient pas de répertoire 'App_Code', ajoutez-en un comme décrit dans la partie précédente.
2. Cliquez ensuite sur le répertoire 'App_Code' puis 'Add New Item...'. Sélectionnez le template 'ADO.NET Entity Model', puis dans l'assistant de création du modèle, sélectionnez 'Generate from Database'.
3. Vous pouvez conserver les paramètres de connexion par défaut proposés par l'assistant puis cliquer sur 'Next'. [Fig.5]
4. Sélectionnez 'Tables' pour générer le modèle de données à partir des tables de la base. Cliquez ensuite sur le bouton 'Finish'. Vous pouvez alors visualiser le modèle de données, les entités, associations... dans Visual Studio 2008 SP1. [Fig.6]

Enregistrer le modèle de données

Il est enfin nécessaire d'enregistrer le contexte de données de l'Entity Framework afin qu'il soit utilisé avec le mécanisme de données dynamiques d'ASP.NET :

1. Ouvrez le fichier 'Global.asax'.
2. Dans la méthode 'RegisterRoutes', ajoutez le code C# suivant (à modifier en fonction de votre base et/ou de votre langage de programmation) :

```
model.RegisterContext(typeof(AdventureWorksLT2008_DataModel.A
dventureWorksLT2008_DataEntities),
    new ContextConfiguration() { ScaffoldAllTables = true });
```

3. Enregistrez puis fermez le fichier 'Global.asax'.

Test du site Web

1. Exécutez l'application Web en mode Debug (touche F5) ou depuis votre navigateur. [Fig.7]
2. Vous pouvez ensuite naviguer dans le modèle de données, cliquer sur les tables, ajouter, supprimer ou modifier des enregistrements, et tout cela avec uniquement 1 ligne de code.

Conclusion

Il y a tellement de nouveautés et d'améliorations dans ces *service packs* 1 de Visual Studio 2008 et .NET Framework 3.5 qu'il serait dommage de ne pas en profiter dès maintenant, en attendant .Net 4.0 en 2009 (ou 2010) ! Et pour conclure de la manière la plus complète possible, nous nous devons de parler très rapidement du *service pack* 1 de Team Foundation Server 2008 : En substance, des améliorations telles que le drag&drop vers l'explorateur de code source, stabilité et performance de l'outil de migration de Visual SourceSafe vers TFS, support de SQL Server 2008, création de projets d'équipe par script, performance accrue du serveur TFS et de nombreux ajouts sur le système des Work Items.

Références

- NET Framework Client Profile: <http://blogs.msdn.com/bclteam/archive/2008/05/21/net-framework-client-profile-justin-van-patten.aspx>
- Le portail MSDN pour les *service packs* 1 de Visual Studio 2008, .NET Framework 3.5 et TFS 2008: <http://msdn.microsoft.com/en-us/vstudio/products/cc533447.aspx>
- Visual Studio 2008 Service Pack Preparation Tool: <http://www.microsoft.com/downloads/details.aspx?FamilyId=A494B0E0-EB07-4FF1-A21C-A4663E456D9D&displaylang=en>
- Une présentation en ligne du .NET Framework 3.5 SP1: <http://www.slideshare.net/ChrisKoenig/net-framework-35-sp1-overview?type=powerpoint>



■ Frédéric QUEUDRET

CTO de la société Mpoware, accélérateur d'innovation. Société française d'édition de logiciels et de prestations de services sur la création d'outils de productivité pour les développeurs Visual Studio. <http://www.mpoware.com/>

Le Model-Driven Testing

Le Model-Driven Testing, plus connu sous le nom de Model-Based Testing (MBT), est une pratique de vérification de la qualité qui permet de générer automatiquement des tests en se basant sur une représentation du système à tester. Les tests ainsi générés permettent de vérifier que le système est conforme au modèle qui le représente, et donc exempt d'anomalie si le modèle est correct par rapport aux spécifications du système.

Il existe deux approches principales pour le MDT : soit la conception interne du système est modélisée (modélisation " architecturale "), soit le comportement externe, celui qui est exposé aux utilisateurs ou aux systèmes interagissant avec le système sous tests, est modélisé (modélisation " comportementale "). La première approche permet en général de réutiliser les modèles de conception ou d'architecture s'ils existent, la deuxième approche nécessite de créer un modèle de toute pièce spécifique au test.

Comment ça marche ? [Fig.1]

La première étape est de réaliser le modèle qui permettra la génération de tests. Le modèle se résume souvent à un ensemble de machines à états, auxquelles on peut adjoindre des informations supplémentaires sous une forme interprétable par l'outil de génération choisi. Les formalismes utilisés pour construire le modèle varient du simple langage de programmation à des notations formelles telles que l'UML.

La deuxième étape est de générer les tests depuis le modèle. Un moteur de tests parcourt le modèle et détermine les cibles de tests. Une cible de test est un objectif à atteindre, par exemple une transition ou un état. Un générateur de tests calcule ensuite les différents chemins pour atteindre ces cibles et les contraintes associées à ces chemins en termes de jeux de données. Certains outils permettent de configurer la génération pour obtenir des tests plus ou moins poussés sur l'atteinte des cibles, par exemple les cas aux limites, les cas d'erreur, etc. A cette étape les tests sont dans un état abstrait, c'est-à-dire dans un format propre à l'outil et

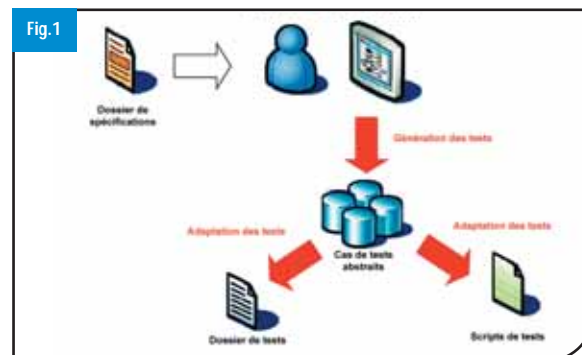
non interprétable par un humain ou un automate de test.

Une adaptation des tests calculés vers un langage cible est donc nécessaire. On note à cette étape deux approches : l'approche online et l'approche offline.

Dans l'approche " online ", l'outil de MDT se connecte au système et exécute les tests via des plug-in dédiés. Dans l'approche " offline " l'outil génère des scripts de tests dans un langage cible. Ce langage cible peut être du simple texte pour une exécution manuelle, ou bien un langage de script pour exécution avec un outil de test automatisé, par exemple QuickTestProfessional. Dans certains cas, il faudra compléter les scripts générés avec des bibliothèques d'actions génériques, par exemple lorsque les tests doivent interagir avec une interface graphique.

Pourquoi utiliser MDT ?

Ce qui apporte de la valeur dans l'approche MDT c'est l'effet de levier sur l'activité d'identification et de création des tests. En effet, l'activité de tests a pour propriété d'être infinie à la différence des autres activités d'ingénierie logicielle. On peut ainsi tester indéfiniment une application et continuer de trouver des anomalies. Il est donc indispensable de se définir une stratégie de tests afin de maîtriser la charge et le délai des tests tout en maximisant le nombre de défauts identifiés. C'est une activité basée sur la gestion du risque que l'on résume le mieux avec le dicton " tester c'est choisir ". Par rapport à une approche classique, l'approche MDT permettra avec la même charge d'identifier et d'exécuter un nombre beaucoup plus important de tests dans un délai beaucoup plus court et donc d'avoir



une stratégie de tests plus efficace. La clé de la réussite d'une approche MDT repose sur l'alignement de l'effort de modélisation avec sa stratégie de test.

*Fabrication
du modèle
de tests*

Où utiliser MDT ?

La démarche MDT est intéressante pour tester la dynamique d'un système ou d'une application, notamment lorsque la combinatoire de cas à tester est très importante. La logique statique (calculs mathématiques, statistiques, etc.) tire moins de bénéfice de cette approche. En effet, les règles de calculs complexes non modélisables doivent être exprimées dans le modèle en un langage formel afin de générer des tests. Cela conduit en fait à dupliquer le code métier dans le modèle, et donc dupliquer les risques d'anomalies. Il faudra alors tester que la règle de calcul exprimée dans le modèle est correcte et sans anomalie. Il est également souhaitable que l'organisation qui met en œuvre du MDT soit confortable avec la production de modèles. En effet, s'il faut changer l'organisation type des projets, les rôles, les compétences, pour introduire cette pratique, cela prendra plus de temps et l'investissement sera plus lourd, au-delà de la simple formation des personnes aux techniques de MDT. La dimension conduite du changement

Fig.2

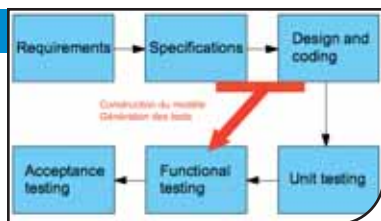
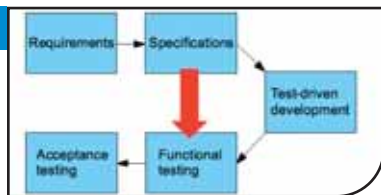


Fig.3



MDT comportemental lors d'un développement dirigé par les tests

est à prendre en compte car c'est une bonne partie du cycle de développement des projets qui change.

Quand utiliser MDT ?

Il faut modéliser le plus tôt possible, en général dès que les spécifications sont disponibles, en parallèle de la conception ou du codage. Dans un cycle itératif on considèrera la construction itérative du modèle en se basant sur les objectifs de l'itération [Fig.2]. La création des modules manquants pour finaliser les scripts automatisés devra néanmoins attendre que l'application soit disponible dans le cas où une interface graphique permet d'interagir avec le système.

Pour quels types de développement ?

L'approche " architecturale " est mieux adaptée à un processus de développement en cascade où la conception est faite avant le code. Le modèle doit en effet rester synchronisé avec la façon dont le code est conçu, ce qui nécessite une maintenance importante dans le cas des processus itératifs et agiles où la conception émerge progressivement.

De plus, cette approche n'est sérieusement envisageable que sur des projets de développement, elle se prête assez mal à des stratégies de " progicalisation " ou des projets d'intégration où l'architecture du système sous test est rarement connue du client final. L'approche " comportementale " s'adapte en revanche à tous types de processus de développement car elle ne nécessite de refactoring du modèle

que lors de la modification des exigences. Elle s'adapte donc bien aux processus agiles où la conception émerge souvent par refactoring de code et n'est donc pas décidée en amont du codage. Il faut également noter que cette approche est complémentaire d'une approche TDD (Test-Driven Development), les tests générés en MDT n'ont pas vocation à remplacer les tests produits et utilisés lors de la pratique de TDD car ils sont d'un niveau différent et participent à un objectif différent. Les tests produits en MDT ont un objectif de vérification et pourront être utilisés lors des étapes de tests fonctionnels, alors que les tests produits en TDD ont un objectif de " micro-spécification " pour les développeurs [Fig.3]. L'approche " comportementale " a également beaucoup de sens dans les projets de " progicalisation " ou d'intégration de systèmes. Sur ce type de projets, le système sous test est une boîte noire que l'on doit tester sous tous les angles et dont on ne connaît pas les détails de conception internes mais seulement le comportement externe qui est décrit dans les spécifications qui font l'objet du contrat entre le client et le fournisseur.

Quelques outils du marché

- **Smartesting** [Fig.4] : cet éditeur français propose Test Designer, une approche MBT basée sur un générateur qui utilise un modèle UML de type comportemental (modélisation du comportement exposé). Le modèle doit être fabriqué avec un outil dédié (Together, RSM,...) en utilisant un sous-ensemble d'UML : les diagrammes de classes, d'objets, et d'états. L'utilisation d'OCL permet de décrire les contraintes non-modélisables. Un addin implanté dans l'outil de modélisation permet de vérifier les règles de modélisation (sorte de compilateur de modèle). Le modèle est ensuite exporté vers l'outil de génération des tests Smartesting Test Designer qui permet de calculer les cibles et de générer les tests. Test Designer fournit un certains nombre d'adaptateurs standard pour générer des tests en format littéraire, ou des scripts pour des outils tels que JUnit ou QuickTestProfessional,

- **Conformiq** : cet éditeur finlandais propose 2 outils de MBT permettant soit une modélisation architecturale, soit une modélisation comportementale. Les deux outils se basent sur des modèles formalisés en UML,
- **Reactive Systems** : cet éditeur propose Reactis, un outil permettant de générer des tests depuis des modèles Simulink et Stateflow,
- **T-VEC** : cet éditeur propose également un outil permettant la génération de tests depuis des modèles Simulink ou Stateflow.

Avantages

- automatisation complète des activités d'identification et création des tests permettant un effet de levier sur la couverture de test
- approche systémique : si le modèle est " complet ", alors la couverture de test sera " complète ",
- traçabilité automatique entre tests et exigences,
- coût de maintenance des scripts de test plus faible que dans une approche classique, il suffit de modifier le modèle et de re-générer les cas de tests,
- la construction d'un modèle de test permet souvent de clarifier les exigences et d'éliminer les écarts de compréhension beaucoup plus tôt dans le cycle de développement

Inconvénients

- L'effort de modélisation est important et ne doit pas être sous-estimé, ainsi pour des diagrammes UML la modélisation nécessite une utilisation avancée des diagrammes d'états et souvent l'utilisation d'OCL pour exprimer des contraintes non modélisables,
- la couverture et la pertinence des tests dépendent exclusivement de celles des modèles fournis à l'outil. Représenter dans un modèle les scénarios de test les plus tordus, ceux qui en général contiennent une bonne dose d'anomalies est une option coûteuse.

Résultat d'une génération avec Smartesting Test Designer

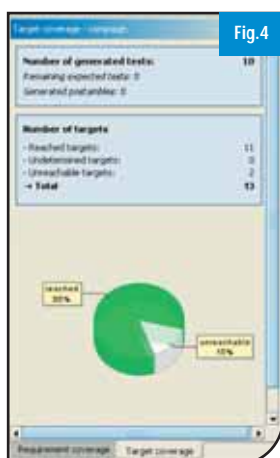


Fig.4



■ Gilles MANTEL
Consultant Senior chez
Valtech Technology
gilles.mantel@valtech.fr

Présentation du module d'extension de Perforce pour Eclipse

Pour travailler avec Perforce dans une interface IDE Eclipse.



Module d'extension de Perforce pour Eclipse

Le module d'extension de Perforce pour Eclipse permet aux développeurs d'accéder facilement au système de GCL Perforce depuis leur interface IDE Eclipse. Il propose les fonctionnalités suivantes :

- Accès rapide à l'historique complet des fichiers
- Prise en charge complète du développement collaboratif, avec possibilité de fusionner les fichiers
- Possibilité de travailler hors ligne lorsque la connexion avec le serveur Perforce est indisponible
- Outil de comparaison des fichiers et prise en charge du suivi des défauts intégrés
- Prise en charge de la fonction de refactoring de l'environnement Eclipse

Le module d'extension de Perforce pour Eclipse prend en charge les systèmes d'exploitation Windows et Linux. Et ce n'est que l'un des nombreux outils intégrés dans le système de GCL Perforce.



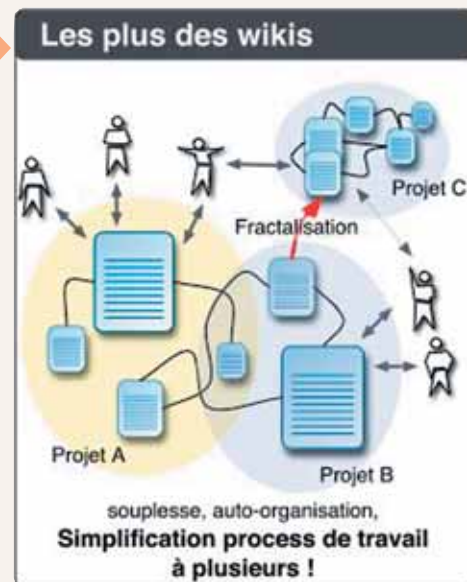
Travailler en équipe :

les outils du succès



Droits réservés

La collaboration au sein des équipes informatiques est devenue un enjeu important pour la réussite des projets. Les outils et les méthodes ne manquent pas ! Comment les utiliser et pour quoi faire ?

Source : Blog Laurent Lunati - www.communitywiki.org/odd/LaurentLunati/CkoUnWiki

avec la complexité des développements et des technologies, les outils de collaboration (liés au développement / développeur) s'intègrent de plus en plus dans les environnements ALM (suite d'outils pour gérer le cycle de vie applicatif), qui nécessitent davantage de collaboration, de partage. Et dans le cadre d'équipes distribuées géographiquement, cela devient particulièrement important ! Encore plus, quand l'équipe comprend des freelances et des développeurs en poste. Car comment garder le contact, comment organiser le développement, le code, le planning, etc. Seuls des outils de collaboration et de partage permettent cela.

Un outillage impressionnant selon les situations

Finalement, qu'est-ce que la collaboration pour un développeur ? Cela recouvre plusieurs réalités : des outils orientés code, des outils de communication, des bonnes pratiques et de la méthodologie. Côté outils, vous avez le choix entre des *standalone* c'est-à-dire prendre un outil pour chaque fonction (gestion du source, serveur, messagerie instantanée, bug tracker, etc.) ou prendre un environnement intégré, ou encore, 3e possibilité, une solution collaborative + ALM. " L'efficacité à travailler en commun est assez inégale. Il faut avoir les bons processus et les bons outils. Il faut " forcer les gens " mais

pas les contraindre. Les outils (seuls) ne sont pas une solution. " analyse Rodolphe Aymard (Itelios).

Cela va dépendre de vos objectifs. Vous n'utiliserez pas les mêmes outils avec quelques développeurs ou des dizaines. Pour une équipe réduite, le choix d'outils *standalone* peut se justifier par le coût et leur efficacité. " Chez The Coding Machine, nous avons une approche axée open source. Nous avons mis en place des forums, cela n'a pas fonctionné, puis nous avons pris de la messagerie instantanée, couplée à du mail pour garder le contact (les développeurs de la SSII sont dans le monde entier et indépendants, ndlr). Pour la gestion du code, nous utilisons Subversion. Au départ, chaque développeur intervenait uniquement dans son répertoire pour aller vers un autre répertoire (d'un autre développeur) il fallait qu'il

demande au chef de projet. Maintenant, chaque développeur peut aller dans les répertoires et modifier le code. Mais avec un tel outil on peut revenir en arrière, corriger si nécessaire ", explique David Négrier (The Coding Machine). Bref, il faut pouvoir doser la liberté que l'on donne aux développeurs dans les accès à l'ensemble du projet.

" Nous travaillons sur 3 sites distants, dont un au Brésil pour nos différents projets, avec un chef de projet à Paris. Nous travaillons beaucoup au forfait et nous mettons en œuvre différents outils de collaboration, en particulier le couple Visual Studio 2008 et Team Foundation Server. Cela nous sert à gérer les sources, les tests, le planning. Nous le couplons à un bugtrack ", commente Rodolphe Aymard.

Quoi qu'il en soit, la collaboration développeur amène à être rigoureux dans la gestion technique et humaine, surtout quand les personnes sont dispersées géographiquement et parlent éventuellement plusieurs langues. La définition des spécifications du projet est donc une étape cruciale. " On découpe le projet en " composants " de petite taille, même si cela dépend des langages et des technologies " poursuit David Négrier. Cette approche permet de faire des itérations régulières, en moyenne toutes les semaines, voire toutes les deux semaines. Une question fondamentale doit se poser. Si je prends une solution tout en un, pourrais-je l'adapter à mes besoins ? En prenant une approche Sourceforge, the coding machine s'est aperçu du manque de souplesse dans l'évolution. Ainsi, la SSII française a développé sa propre solution en interne, à travers Subversion, pour pouvoir intégrer l'ensemble des éléments nécessaires au projet, à la gestion projet et à la collabora-

tion. Une sorte de CRM de développement ! Un autre point à ne pas négliger : la complexité des solutions intégrées. Nous l'avions déjà noté dans le numéro de septembre de Programmez ! à propos de la plate-forme Jazz et de son implémentation Rational Team Concert, particulièrement complète, mais parfois complexe à manœuvrer. " Team Foundation Server est assez simple mais il possède de nombreux modules. Il ne faut pas tout déployer d'un bloc mais se l'approprier au fur et à mesure " précise Rodolphe Aymard.

Intrusion ou non intrusion ?

Il faut effectivement un point d'équilibre. Si un outil est trop intrusif, le développeur perd en productivité, s'il ne l'est pas assez, le risque est de perdre un certain contrôle du projet, du développement. Mais ensuite, c'est au chef de projet à trouver les règles de fonctionnement.

Mais les outils seuls ne suffisent pas. Il faut aussi des méthodes, des bonnes pratiques. Ainsi, Itelios s'inspire de CMMi pour mieux organiser les relations chefs de projet, clients, développeurs. Par contre la méthode XP ne sera pas adaptée à toutes les situations et la complémentarité de la paire de développeurs sera primordiale. La relation avec le client est un élément sensible. " Nous avons envoyé des développeurs en formation commerciale " précise R. Aymard ! Le développeur voit ainsi ce qui se passe de l'autre côté, et prend conscience qu'il n'est pas seul. Un développeur sera d'autant plus sensible à la collaboration s'il collabore lui-même à des projets open source, communautaires.

■ François Tonic

La collaboration web 2.0 selon les analyses de Forrester

" Les technologies collaboratives Web 2.0 apportent des solutions aux problèmes auxquels sont actuellement confrontées les entreprises, mais la majorité de celles-ci n'utilise à aucun moment de façon optimale le potentiel de ces outils ", déclare Gil Yehuda, analyste pour Forrester Research.

" Cette nouvelle recherche montre aux entreprises où investir intelligemment et où miser stratégiquement. Au vu du contexte économique actuel, Forrester pense que les outils de collaboration peuvent permettre aux entreprises de réduire leurs coûts d'exploitation en rassemblant les individus et les processus de manière rapide et efficace. "

Forrester prévoit que les technologies collaboratives suivantes de l'entreprise 2.0 continueront à se développer :

Les réseaux sociaux transformeront la nature du travail. Les réseaux sociaux fournissent du contexte et du contenu. Une résistance culturelle existe, mais d'après Forrester, elle finira par céder, ce qui permettra aux employés de se rapprocher de leurs pairs afin de créer une collaboration qui n'existait pas auparavant dans l'entreprise.

Les wikis contribuent à transformer la collaboration. C'est l'une des technologies Web 2.0 les plus prometteuses pour l'entreprise. Les utilisateurs déclarent avoir du succès avec leurs projets wikis, notamment lorsqu'ils sont sponsorisés par des sociétés leaders et intégrés dans des processus d'entreprise - et ce marché montre des signes forts de croissance.

Les blogs ne sont pas en reste, mais ils ne captent ni ne retiennent l'attention des employés d'une entreprise. Les réseaux sociaux apportent une seconde vie aux blogs internes en offrant plus de contexte au contenu des blogs, mais d'après Forrester, les blogs seuls n'attirent pas l'attention des employés.

Les flux RSS sont sous-estimés au sein de l'entreprise. Cette technologie omniprésente propose un mécanisme permettant de fournir du contenu aux individus selon leurs besoins, plutôt que de les laisser faire la recherche eux-mêmes.

Nouvelle rubrique

+ de **4000** offres d'emploi
www.programmez.com

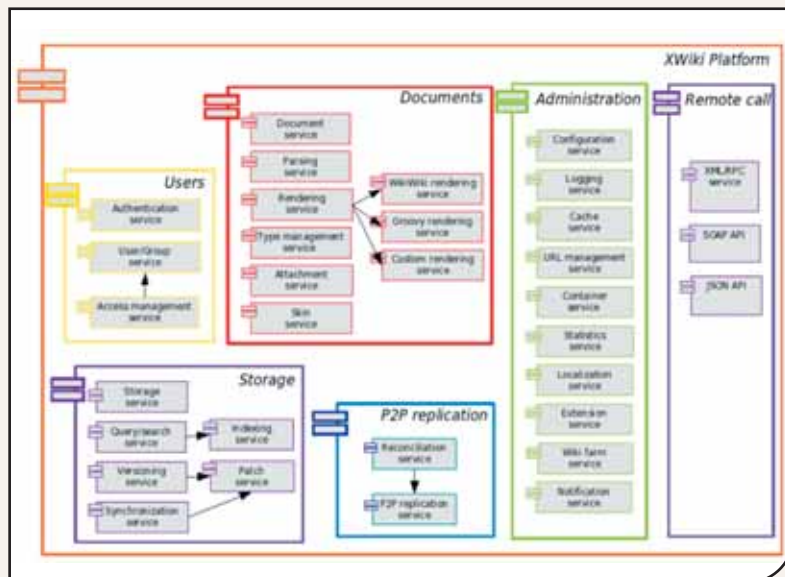
Wiki et les développeurs



Sur le CD
du numéro 1



De nombreuses communautés de développeurs utilisent le wiki pour soutenir leur processus de développement, aussi bien dans le monde de l'Open Source que chez les éditeurs traditionnels. Un wiki est un logiciel permettant de créer un site web sur lequel les utilisateurs autorisés vont avoir la possibilité d'éditer les pages et d'ajouter du contenu très facilement. Il permet donc à une équipe de partager des informations et de collaborer facilement et rapidement.



Depuis l'apparition des wikis en 1995, leurs fonctionnalités basiques ont relativement peu évolué (édition facile des pages et historique des modifications, création de liens). On a ensuite progressivement assisté à l'apparition de wikis offrant beaucoup d'options en termes d'extensibilité et de modularité. TWiki est créé en 1998 avec l'objectif de fournir des services tournés vers l'entreprise. XWiki voit le jour en 2003 et offre la possibilité de créer des applications (blog, FAQ) au sein même du wiki. Dans cet article, nous allons voir les avantages apportés par la création d'applications par rapport à un wiki traditionnel.

Pourquoi créer une application dans son wiki?

La force du wiki se trouve dans la simplicité avec laquelle elle permet à ses utilisateurs de modifier les contenus existants. Elle s'oppose notamment aux templates de structuration offerts par les CMS traditionnels. Cependant, il est parfois intéressant de pouvoir combiner contenus structurés et peu structurés sur une même plate-forme. C'est ce que permettent de faire les applications. Une application permet de définir un type de document donné (par exemple une fiche produit avec un titre, un champ

de description, une image et un tableau avec ses prix) et de créer de nouvelles pages qui utilisent ce modèle. Il est ensuite possible de lister toutes les pages qui utilisent un modèle donné ("Liste de toutes les fiches produit disponibles sur ce wiki") à l'aide d'un langage de requête.

De ce fait, il devient possible de combiner sur la même plate-forme des données structurées et faciles à exploiter (les informations présentes dans les fiches produit) et des données non-structurées (pages wiki traditionnelles). Cela offre plus de souplesse et de flexibilité que les approches traditionnelles. Dans cet article, nous allons décrire brièvement comment il est possible de créer une petite application à partir de la plate-forme XWiki.

Quelques considérations techniques

Architecture client-serveur

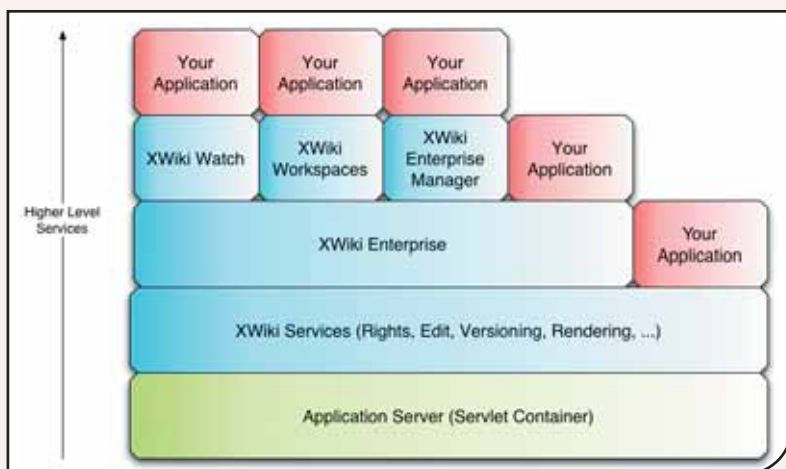
Un wiki fonctionne suivant une architecture client-serveur traditionnelle. Le logiciel tourne sur un serveur distant connecté à internet, et ses utilisateurs accèdent au site et à ses contenus à travers leur navigateur web. Cela implique qu'il est possible de se connecter à un wiki et de modifier ses contenus depuis toute machine connectée à internet. Certains

wikis offrent d'autres moyens d'accéder à leur contenu au travers de leur API (REST, XML/RPC, RSS), ce qui permet parfois l'utilisation d'un client riche pour interagir avec le wiki. Par exemple, le logiciel XEclipse (basé sur Eclipse RCP) permet d'éditer les contenus d'un XWiki Enterprise (une solution de wiki professionnelle) en mode déconnecté et de les synchroniser avec les contenus du wiki lors du retour en mode connecté.

Couches applicatives

Le wiki en tant que logiciel est le plus souvent composé de plusieurs couches applicatives superposées. Dans le cas de XWiki Enterprise, on peut ainsi compter :

- **Le serveur d'applications** : XWiki Enterprise étant écrit en Java, il nécessite une JVM et un serveur d'applications pour tourner. Dans le cas de XWiki Enterprise, ce peut être Jetty, Glassfish, JBoss ou encore Apache Tomcat.
- **La base de données** : c'est là que sont stockés les contenus du wiki à proprement parler. XWiki Enterprise utilise Hibernate, qui permet la connexion à un grand nombre de bases de données parmi lesquelles on citera MySQL, Derby, Oracle, PostgreSQL ou encore HSQLDB.
- **La business logic** : c'est ce qui constitue le coeur du wiki à propre-



ment parler : gestion des documents (édition, historique) et des utilisateurs (droits d'accès), fonctionnalités d'import et d'export. Elle est constituée de composants qui s'interfaçent entre eux et est écrite en Java. Elle peut être étendue à l'aide de nouveaux composants, et s'appuie sur de nombreux composants Open-Source (Struts...)

- **Les applications** : cette couche est particulière à XWiki Enterprise. C'est ce qui fait de cette solution un wiki de seconde génération. A l'aide des langages de script Velocity et Groovy et d'un système de classes et d'objets, elle permet la création d'applications (telles qu'un blog ou un album photo) au sein même du wiki.
- **L'interface utilisateur** : c'est ce que l'utilisateur voit lorsqu'il se connecte sur le wiki. Elle utilise les technologies web standard (HTML, CSS, JavaScript).
- **Le serveur web** : il permet de servir les pages du wiki en ligne et de gérer les connexions et le cache. Le plus communément utilisé est Apache HTTP.

Créer une application

XWiki utilise un modèle de classes et d'objets pour définir le modèle de données d'une application. Il fonctionne de la façon suivante

(nota : la longueur de cette article ne permettrait pas d'inclure un tutorial complet sur la création d'applications dans Xwiki. Vous trouverez un lien vers un tutorial complet en fin d'article.)

- **Le développeur définit une classe.** Une classe est constituée d'un ensemble de propriétés (string, number, textarea...). Ainsi, dans le

cas d'une fiche produit, on définira une propriété de type *string* pour stocker le nom du produit, une propriété de type *textarea* pour stocker sa description. D'une façon plus ludique, on peut voir la classe comme un moule à gâteaux.

- **Le développeur pourra ensuite créer des objets à partir de cette classe.** Un objet est une instance d'une classe donnée. Par exemple, les données relatives à une fiche produit seront stockées dans l'objet correspondant. Chaque objet peut être considéré comme un gâteau fait à partir du moule initial.
- **Il faut aussi définir une ClassSheet pour chaque classe.** Le template définit la façon dont les propriétés de la classe vont être affichées au sein d'une page wiki. Par exemple, c'est dans la ClassSheet que l'on définit que le nom du produit sera affiché en gros caractères en haut à gauche de la page, sa description en petits caractères en dessous, son image de référence sur la droite de la page.
- **Enfin, il faut définir le ClassTemplate de cette classe.** Ce template est la page de référence qui sera copiée à chaque fois qu'un nouveau document contenant un objet de cette classe est créé. Le template contient un objet vide de la classe choisie. Ainsi, lorsqu'on créera une nouvelle fiche produit, elle le sera selon le modèle défini dans le template " Fiche Produit ".

Une fois ces étapes suivies, le modèle de données est prêt : les utilisateurs vont pouvoir créer de nouveaux documents qui suivent le modèle de page défini dans la classe. Pour cela, il suf-

fit d'ajouter un formulaire sur la page d'accueil de l'espace dans lequel se trouve votre application. Vous pouvez également ajouter une recherche dans les documents correspondant à l'application.

Au delà du système de classes et d'objets, XWiki possède une API très riche au travers de laquelle il est possible d'accéder à un grand nombre d'informations : par exemple, insérer `$doc.creationDate()` dans une page affichera la date à laquelle ce document a été créé. `$doc.author` affichera le nom de la dernière personne à avoir édité le document en question. De plus, XWiki offre la possibilité d'utiliser des langages de templating et de scripting au sein même du wiki :

- **Velocity** : ce langage de templating simple à manier peut être utilisé pour opérer des actions simples sur vos pages. Par exemple :

```
#if($context.user == " XWiki.Admin ")
```

 Vous êtes l'administrateur par défaut de ce wiki !

```
#else
```

 Vous êtes un utilisateur classique.

```
#end
```

 affichera un contenu différent selon que l'utilisateur en train de consulter la page est l'administrateur par défaut du wiki ou pas.
- **Groovy** : ce langage de script qui étend Java a vocation à permettre le traitement de tâches plus complexes (telles que l'import d'un fichier XLS) au sein même d'une page wiki. Il requiert l'utilisation d'un droit de programmation spécifique.

Pour en savoir plus

- <http://www.xwiki.org/> : le site web de la communauté open-source XWiki
- <http://www.xwiki.org/xwiki/bin/view/Main/Download> : la page de téléchargement de XWiki
- <http://platform.xwiki.org/xwiki/bin/view/DevGuide/Tutorials> : la page listant les tutoriels permettant de démarrer avec XWiki
- <http://www.theserverside.com/tt/articles/article.tss?l=XWiki> : un article décrivant en détail la création d'une application dans Xwiki



■ Guillaume Lerouge
Product manager chez
XWiki SAS, éditeur de
solutions de collabora-
tion wiki.



Les outils de collaboration

Mission impossible de dresser un panorama des outils de développement. Il en existe de toutes sortes, plus ou moins étendus dans les fonctionnalités. Aujourd'hui, on trouve plus de 200 projets et solutions autour de la collaboration ! Le choix ne manque pas. Voici quelques pistes pour y voir plus clair.

Les wiki

Mindtouch deki : environnement wiki, avec des éditions open source et commerciales. Deki propose de très nombreuses fonctions comme une librairie de développement, support des flux RSS, intégration avec Outlook, connecteur à des CRM, ERP. Il peut aussi générer des rapports. Son architecture est orientée web. Il s'intègre avec les logiciels de contrôle de code source et de bugs comme par exemple Subversion, Mantis, Trac, et Jira.

Xerox Codendi : basé sur les outils open source reconnus, Codendi est une plate-forme collaborative globale gérant le code source, le projet, l'agenda, les espaces personnels, le multi projet, les librairies composants, la communication dans les équipes, etc. Il s'interface facilement avec Eclipse. On peut mettre en place un système de sauvegarde et une connexion avec son annuaire. Sur la communication, on a le choix entre les mailing lists, les forums, les annonces, les flux rss, etc.



Xwiki : projet générique de wiki, français. Il s'intègre facilement à l'environnement entreprise, possède des outils d'administration et une plate-forme complète pour mettre en place ces wiki internes ou externes. Il est possible de personnaliser xwiki.

IBM avec Jazz et Team Concert

Dernier venu dans le monde de la collaboration, Jazz est une plate-forme technologique que les éditeurs peuvent intégrer dans leur solution. Orienté web, Jazz repose sur OSGi et le client peut être web ou lourd. Fondamentalement, Jazz repose sur trois piliers : la collaboration (fourniture de work items, de flux de collaboration, messagerie instantanée, etc.), gouvernance (utilisation de méthodes agiles, définition des processus de gouvernance) et productivité (suivi des tâches, planification). IBM dispose d'une première implémentation avec Rational Team Concert, à terme, Jazz sera présent dans l'ensemble des outils Rational. 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. 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).

Les solutions Microsoft

Groove

Microsoft Office Groove 2007 est une application sécurisée de travail collaboratif, particulièrement adaptée aux équipes mobiles ou géographiquement distribuées. Elle permet de relier très simplement des interlocuteurs issus d'une même entreprise ou d'organisations différentes, et leur offre la capacité d'échanger des informations formelles ou informelles au sein d'espaces de

travail cloisonnés et sécurisés. Groove propose en standard un certain nombre d'outils que l'on peut librement ajouter dans les espaces de travail: stockage de fichiers, forum de discussion, calendriers, formulaires, suivi de réunions. Un puissant outil largement méconnu !

Team System

Microsoft Team System est un environnement de collaboration et de gestion de projets. Il intègre de très nombreuses fonctions comme les tests, la gestion du code, le suivi de projet, la génération de rapport, la configuration, les spécifications, etc. La partie serveur intègre Team Builder pour le build projet. Il peut aussi générer un site web projet via sharepoint. On peut se connecter à un serveur Team System Server soit avec Visual Studio, soit Eclipse. La prochaine version majeure est Rosario.

Quelques autres solutions

Novell Teaming : Teaming est une solution de collaboration d'entreprise pour les équipes. Il intègre les blogs, la messagerie instantanée, la gestion documentaire, le protocole webDav, le support LDAP, la gestion des portails. On peut disposer de fonctions de conférences, gestion calendaire, etc.

Unity Team Developer : environnement orienté développement, il permet de gérer le cycle de vie des applications. On peut définir les tâches, gérer les projets et les sources, les versions, les modèles de données, etc. Il intègre un outil RAD, un moteur de reporting et de nombreux connecteurs.

Perforce propose une solution autour de la configuration et de la gestion du code. Même si cela n'est pas de la collaboration au sens premier du terme, les outils de gestion de configuration et de projet sont à la base des environnements de développement pour le développeur. L'autre référence du marché est Subversion, qui remplace avantageusement CVS.

■ François Tonic



W4-LYRIA : la modélisation au coeur des applications métier



ihm

EN TOUTE SIMPLICITÉ !

Pour implémenter des applications riches, accédant à des données hétérogènes (applicatifs métier, BD, et depuis la V4.1, Hibernate, GED et silos documentaires), déployées en Swing, plugin Eclipse ou DHTML/Ajax et s'insérant naturellement dans les processus d'entreprise...

Adoptez la puissance et l'agilité de l'approche Model-Driven

**NOUVELLE
VERSION
LEONARDI
V4.1
open source**

Concentrez-vous sur votre métier et dotez votre entreprise d'avantages compétitifs durables: amélioration du cycle de vie du logiciel, démarche itérative par prototypage pour coller aux besoins, découplage technologie/métier, évolutivité, agilité et évolutivité, le tout sans expertise technique pointue !



Pour en savoir plus sur la solution LeonardI, rendez-vous sur notre site www.lyria.com ou envoyez-nous un courriel à info@lyria.com



EPITA, l'école d'ingénieurs informaticiens

Près de 25 ans après sa création, l'Epita recueille les fruits de sa persévérance : la reconnaissance par l'Etat et la CTI (Commission des Titres d'Ingénieur).

Joël Courtois, directeur de l'Epita
" Plutôt que de prendre les meilleurs, l'Epita veut prendre ceux qui peuvent devenir les meilleurs. "

Après avoir gagné la reconnaissance comme école d'ingénieurs en 2007, l'Epita est depuis cette année reconnue par l'Etat. Ses dirigeants soulignent qu'elle se distingue ainsi des autres écoles qui forment des informaticiens à bac + 5, mais ne délivrent pas le titre d'ingénieur diplômé. De plus, de par sa reconnaissance par l'Etat, elle est désormais autorisée à accueillir des étudiants boursiers de l'enseignement supérieur.

Le titre d'ingénieur : avantage ou contrainte ?

" Le titre d'ingénieur était une demande des familles ", indique Joël Courtois, directeur de l'Epita. Le dossier CTI (Commission des Titres d'Ingénieur) a été déposé en 2000. Son obtention, 7 ans plus tard, a été conditionnée par plusieurs facteurs : pour mériter ce titre, l'école doit faire état de coopérations au niveau international, développer des activités de recherche, collaborer avec l'industrie, se conformer à des normes de pro-

grès (académique, enseignement, recherche). C'est le prix à payer pour la reconnaissance du diplôme par les employeurs. " *L'impact CTI joue surtout sur les grosses structures qui ont des grilles précises* ", poursuit Joël Courtois. Même si, pour la plupart des sociétés, la réputation de l'Epita est généralement suffisante.

La CTI attire des candidatures, mais le nombre de diplômes délivrés est limité à 300 par an. " *Depuis que nous avons la CTI, nous avons reçu le même nombre de dossiers de candidatures que les années précédentes, mais auparavant 1/3 des élèves ne venaient pas à l'entretien* ", constate Joël Courtois. Depuis 2007, l'Epita reçoit 50% d'élèves de plus. En 2007, 217 élèves ont intégré l'Epita. En 2008, 307 ont été admis en première année du cycle préparatoire (Sup), dont 20% ne vont pas passer en deuxième année (Spé). " *Nous complétons en troisième année (première année du cycle d'ingénieur) par ceux qui viennent de prépas classiques.* "

Un ascenseur social

L'Epita est une école privée, donc payante : 6000 € par an en cycle préparatoire, 8000 € par an pour les trois années suivantes, soit 36 000 € pour la totalité de la formation. Cela pose évidemment un problème à ceux qui n'ont pas de ressources et qui viennent de province. Du fait de la reconnaissance par l'Etat, les élèves de l'Epita bénéficient de l'ensemble des œuvres du CROUS et en particulier des bourses d'Etat. Par ailleurs, ils peuvent demander un prêt bancaire garanti par l'Etat (plus de la moitié des élèves de l'Epita empruntent. " *Les élèves sont fiers de financer leurs études, de prendre leurs responsabilités* ", note Joël Courtois). La difficulté concerne surtout les trois premières années, puisqu'en 4e et 5e année les stages permettent de financer les études. " *La reconnaissance par l'Etat nous permet de jouer l'ascenseur social* ", souligne Joël Courtois.

A condition d'arriver au bout du cursus, le retour sur investissement est rapide, puisque le salaire moyen des jeunes diplômés Epita est supérieur à 36 000 €.

Les élèves de l'Epita ont 30 heures de cours par semaine, mais ils passent le plus clair de leur temps dans l'école, qui dispose de salles d'ordinateurs ouvertes 7 jours sur 7, 24 heures sur 24.



D'ailleurs, 70 à 80% des élèves voient leur stage de 5e année transformé en CDI avant la fin des études. Si la formation est bien reconnue par le marché – les jeunes diplômés sont pour une large majorité recrutés avant même la fin de leurs études –, le cursus est " très dur " de l'avis de la plupart des 1000 élèves. " Les étudiants de l'Epita ont la réputation d'être de vrais bosseurs, formés à la réalité de l'entreprise car le rythme de l'école est vif ", assure Jean-Christophe Monier, ancien élève (promotion 93). " Plutôt que de prendre les meilleurs, je veux prendre ceux qui peuvent devenir les meilleurs ", affirme Joël Courtois. D'où la rigueur de la sélection à l'entrée, qui comprend 5 entretiens et un test QCM de culture générale.

■ Claire Rémy

L'Epita en chiffres

- Création en 1984
- 1000 élèves, 30 enseignants chercheurs
- plus de 4500 diplômés à ce jour
- 2 ans de cycle préparatoire intégré (Sup + Spé)
- projet professionnel à la fin du cycle prépa
- 3 ans de cycle d'ingénieur
- 7 options de spécialisation, 1 option recherche 4 options d'orientation
- 5e année de Master of Science in Information Systems
- 25% environ des cours se font en anglais à partir de la 4e année
- 100% d'expérience internationale (au moins 6 mois à l'étranger)



Une nouvelle façon d'enseigner

Selon Fabrice Bardèche,
Vice Président du groupe IONIS

/// Dans le domaine pédagogique, nous sommes dans une période d'évolution majeure. Auparavant, le professeur était le garant de la connaissance. Aujourd'hui, plus rien n'est circonscrit. Les concepts sont remis en cause et on peut très légitimement s'interroger sur leur validité. Internet a tout changé : il faut faire entrer le monde extérieur dans le cours. Le MIT (Massachusetts Institute of Technology) a mis gratuitement tous ses cours en ligne, le professeur n'est donc plus le vecteur exclusif du savoir. Les portes ouvertes sur les savoirs sont tellement nombreuses, les masses de connaissances tellement importantes, que l'on a besoin d'un pilote. C'est justement le rôle du professeur. Les professeurs sont hors des murs, soumis à la critique des autres. Le professeur doit être un bon pilote, guider les élèves dans les savoirs, leur montrer comment les utiliser, comment les architecturer, il montre le cap. C'est sa valeur ajoutée. Il ne faut plus assommer l'élève avec des cours, il faut l'encadrer. On est dans un savoir tellement mouvant : il faut apprendre à apprendre. Cette réflexion pédagogique est valable pour toutes les écoles. "

Deux écoles complémentaires au sein du même groupe

L'Epita et l'Epitech sont les deux écoles spécialisées en informatique du groupe Ionis. " L'informatique est devenue un ensemble divers et complexe, d'où le besoin de se spécialiser en informatique pendant 5 ans ", explique Fabrice Bardèche, vice-président exécutif du groupe. C'est pourquoi ces écoles font de l'informatique un enseignement à part entière, en couvrant tous ses domaines, à l'inverse des grandes écoles scientifiques, ciblées sur une partie de l'informatique. " Les deux écoles forment l'esprit différemment ", poursuit Fabrice Bardèche. " En tant qu'école d'ingénieurs, l'Epita forme des scientifiques, alors que l'Epitech s'adresse surtout à des passionnés. "

Le champ de l'Epita est ainsi défini par Fabrice Bardèche : " C'est l'école de l'intelligence informatique : on y apprend la culture scientifique globale, à resituer ses compétences techniques, ainsi que les connaissances managériales. La spécialisation se fait en dernière année. Elle comporte une réflexion sociétale, économique. "

Pour Joël Courtois, l'appartenance au groupe permet de se mesurer aux grandes institutions étrangères. " Avec 1000 élèves, l'Epita ne fait pas le poids devant l'université de Nanjing, par exemple, qui compte 20 000 élèves sur 4 ans. En revanche, si l'on prend en compte le groupe Ionis, il totalise près de 15 000 élèves et 50 000 anciens ", note le directeur de l'Epita.



Transformer des ingénieurs en consultants métier Finance

Softeam, société de conseils et de formation, spécialiste des technologies objets et des nouvelles architectures logicielles, lance un programme de formation ambitieux pour faire de ses collaborateurs techniques des experts en Finance de Marché.

Softeam s'appuie sur First Finance, organisme de formation en techniques financières, pour monter un Master-Class en Finance de Marché, cursus de 3 mois aboutissant après un examen de fin de cursus, à une certification reconnue sur le marché.

First Finance forme ses consultants et ingénieurs afin qu'ils intègrent les spécificités propres aux métiers de leurs clients. Ainsi, ils pourront compléter leur expertise technologique par la compréhension de la logique commerciale, réglementaire, métier et organisationnelle de leurs clients. Cet élément est un critère important générateur de meilleure productivité et de pertinence.

Master " First Finance "

Le groupe First Finance est le leader européen de la formation à la Finan-

ce. Depuis sa création, il a formé plus de 30 000 professionnels et compte parmi ses clients plus de 250 institutions financières. Il a développé un cursus sur mesure pour Softeam sous forme de " présentiels " et de e-learning à distance. A l'issue du cursus, les collaborateurs formés devront valider leurs connaissances au travers de tests pour accéder à la certification délivrée par First Finance (label reconnu par l'ensemble des professionnels de la banque et de l'assurance). SOFTEAM en est déjà à la 4e promotion de son Master Class First Finance " introduction aux marchés financiers ". Chaque promotion regroupe 12 collaborateurs de SOFTEAM. A terme SOFTEAM aura formé les 130 ingénieurs intervenant dans le secteur métier Finance - Banque - Assurance, et continuera à former ses nouveaux embauchés.



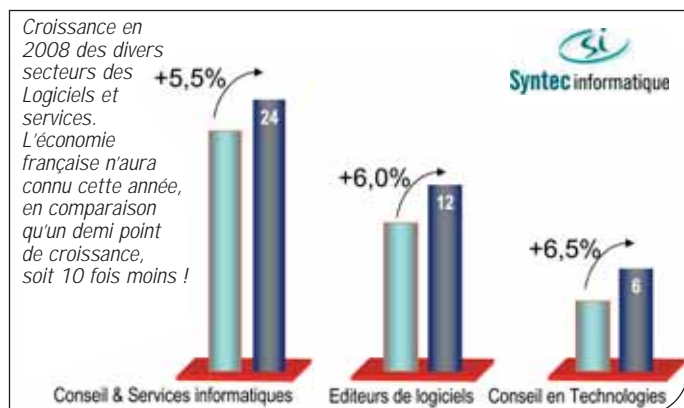
" J'ai acquis un bon socle de compétences "

Le cursus est étalé sur 3 mois. Loïc a participé à la première promotion 2007 :

"Ma mission s'est déroulée au sein d'une grande banque d'investissement et j'avais besoin d'acquérir des connaissances plus larges et approfondies en finance de marché. Cette formation s'est effectuée en deux parties : une partie en e-learning, qui offre une flexibilité appréciable, l'autre partie en trois journées de formation intensive, une chaque mois, animées par des intervenants, anciens traders issus du métier de la Finance, très compétents dans leur domaine. Avec un recul de quelques mois, je considère avoir acquis un bon socle de compétences en finance de marché. J'ai maintenant une vision plus précise de tout ce qui touche au pricing et une meilleure compréhension des informations traitées dans les applications sur lesquelles je travaille."

Retour d'expérience

Marché : 2008 : +6% de croissance pour les éditeurs de logiciels



Le syndicat Syntec informatique affine ses chiffres. Par rapport à ses prévisions publiées en mars 2008, il reste confiant pour le chiffre d'affaires de l'activité Conseil et intégration. En revanche, les Editeurs de logiciels perdent 1 point, leur prévision de croissance passant de 7% à 6%

" Le marché français des Logiciels & Services devrait enregistrer en 2008 une 4e année consécutive de croissance supérieure ou égale à 6%, et atteindre 42 milliards d'euros. Le marché, qui a doublé en 10 ans et quintuplé en 20 ans, est devenu stratégique pour l'économie française ", rappelle le Syntec Informatique

2009 : les secteurs qui auront le vent en poupe

Le syntec prévoit pour 2009 que " les projets métiers, liés aux lancements de nouveaux produits ou services, après une phase d'attente de quelques mois dans certains secteurs, devraient rester nombreux en 2009 ". Les secteurs dynamiques seraient : • Grands projets d'Etat • Dématérialisation, • Décisionnel / Pilotage de l'entreprise • Informatique embarquée • Mobilité.

Syntec est optimiste pour l'externalisation- infogérance : " en premier plan au 1er semestre, malgré des pressions importantes sur les prix ". Enfin, le syndicat prévoit que : " Les besoins réglementaires, notamment dans la Banque, de même que la nécessité de rationaliser les systèmes d'information, comme dans le cadre des rapprochements d'entreprises, devraient représenter des relais de croissance majeurs " En savoir plus : www.syntec-informatique.fr

■ Jean Kaminsky

L'agenda 2009 de cet informaticien est déjà bien rempli



Salon RECRUTEMENT Lesjeudis.com

INFORMATIQUE : DEVELOPPEMENT - AMOA - FINANCE

29 janvier - PARIS

Au Palais des Congrès - de 11h à 21h

15 février - LILLE

5 mars - NANTES

26 mars - GENEVE

2 avril - AIX-EN-PROVENCE

Mai - PARIS

Juin - TOULOUSE

Septembre - PARIS

Septembre - BORDEAUX

Octobre - RENNES

Novembre - BRUXELLES

Novembre - LYON

Renseignements et pré-inscription sur **www.lesjeudis.com**

php

par la pratique !



PHP n'est pas un outil marketing dont les fonctionnalités évoluent en fonction du buzz, PHP est une technologie écrite par des développeurs pour des développeurs. Notre approche est concrète : nous observons la direction que prend le Web, et nous nous assurons que PHP est capable de relever ces défis. Lorsque nous avons lancé PHP 5, il y a maintenant plus de cinq ans, nous avons senti que le Web aurait besoin d'industrialiser les services Web, ce qui nous a donné l'impulsion nécessaire pour moderniser notre gestion de XML, SOAP et des technologies associées. Avec la croissance du Web en Asie, notamment le boom de l'Internet en Chine, il nous est devenu évident qu'un support plus complet de l'internationalisation et de la localisation devenait une priorité. Nous avons lancé

“ Une technologie écrite par des développeurs pour des développeurs ”

le dossier PHP 6 dont la principale nouveauté est la mise à niveau générale de l'Unicode à travers toute la plate-forme.

Outre ces changements dans l'environnement Web lui-même, nous avons vu des évolutions dans l'expérience-utilisateur et dans les attentes de ces derniers : elles ont été bien au-delà de ce qui était attendu de la technologie dans les années 90. Avec de plus en plus d'utilisateurs dans la communauté PHP, qui amènent leur expérience de programmation-objet issue des autres langages, nous avons étendu les fonctionnalités les plus utiles de la POO. PHP ne cherche pas à changer la manière de programmer de ses utilisateurs. Il tend à fournir un outil familier et pragmatique pour résoudre les problèmes Web, et notre objectif à travers l'évolution du langage est de répondre aux attentes les plus courantes. La POO est un

domaine où nous poursuivons une évolution continue.

A ce stade, l'avenir de PHP au-delà de l'horizon de PHP 6 est encore un mystère. Il dépendra directement de la direction que prendra le Web d'ici les cinq prochaines

années. Lorsque des tendances de fond se dessineront, vous pouvez être sûr que PHP les aura constatées, et saura accompagner leur progression.

■ Rasmus Lerdorf, créateur de PHP,

Traduit par Damien Seguy (AlterWay Consulting)

Panorama des IDE

Sur le CD du numéro !



Quand on développe professionnellement, ou que l'on est un utilisateur exigeant de PHP, il est nécessaire d'utiliser le bon outillage pour gérer le grand nombre de fichiers contenant les classes et bibliothèques. Mais l'outil ne faisant pas l'ouvrier, un mauvais outil peut contrarier votre développement !

Au delà des bloc-notes améliorés et simples éditeurs de code avec plugin, intéressons-nous aux EDI (Environnement de Développement Intégrés), qui permettent, en plus du code et des fichiers, de gérer un projet complet, avec bibliothèques. PHP est opensource, de ce fait aucun éditeur de code n'est livré avec lui, c'est au développeur à choisir son outil.

Qu'attend-on d'un EDI pour PHP ?

Un EDI doit pouvoir **gérer mon code** et m'aider à l'écrire : aide à la saisie, complétion des fonctions natives, complétion de mes variables, colorisation, détection des erreurs de syntaxe. Les fonctionnalités très appréciées sont de pouvoir replier mon code, afin de mettre en valeur la partie sur laquelle je travaille. Il est nécessaire de bénéficier d'un débogueur intégré, qui permette de placer des points d'arrêts et de lancer mes scripts depuis mon outil. Un bon EDI doit aussi me permettre de **gérer tout mon projet** : renouveler ou détecter les mauvais chemins d'inclusion, proposer un couplage avec un outil de versionning, me permettre de gérer divers environnements (dev, test, prod), voire un éventuel déploiement (cela peut aller du logiciel FTP à la synchronisation de fichiers). Procédure de test des EDI utilisée pour ce comparatif : construction d'un mini-projet avec deux classes, et reprise d'un projet existant utilisant un framework. Comme tous les développeurs PHP connaissent **Zend Studio**, et savent qu'il intègre un débogueur, un profiler et s'intègre parfaitement avec les outils Zend (à commencer par Zend Framework), j'ai pris le parti de présenter des alternatives, plus ou moins complètes.

Komodo IDE

Licence : commerciale (245 euro)
Notez qu'il existe l'éditeur de code Komodo Edit, gratuit, mais qui ne propose pas de debugger, ni d'explorateur de code. Editeur : ActiveState (<http://www.activestate.com>)

Komodo IDE est très intéressant pour les autodidactes, et débutants, puisqu'il propose de l'aide, tout en finesse. Les développeurs confirmés apprécieront aussi les riches fonctionnalités.

Points forts :

- ajout d'un nouveau fichier : le choix des templates
- suggestion de codage : il précise ce que l'on est en train de faire [Fig.1]
- possibilité de créer directement un projet pour Zend Framework, mais c'est le seul framework proposé
- Quand on efface une ressource, le choix est donné de juste l'ôter du projet ou de supprimer réellement la ressource.
- Assistant pour configurer son debugger

Points faibles :

- ajouter juste un nouveau fichier au projet est un peu caché dans les sous-menus
- support de certains outils (Zend Framework) mais pas tous
- détail : les numéros de lignes ne sont pas visibles par défaut

Delphi for PHP

Licence : commerciale
Editeur : CodeGear (<http://www.codegear.com/products/delphi/php>)

Delphi for PHP n'est pas facile à prendre en main pour les développeurs venant de PHP, mais pourra satisfaire certainement des développeurs venant d'autres langages, plus orientés composants.

Points forts :

- orienté RAD, il propose à l'ouverture de créer une application, ou un composant

- Il propose notamment de nombreux assistants (ou wizard) permettant de choisir des options pour lancer son projet

Points faibles :

- pas de version française
- n'assiste pas au code, par défaut (seconde parenthèse ou accolade, par exemple)
- propose de sauver dans les fichiers, en utilisant les boîtes de dialogue de l'OS
- les formulaires générés sont en HTML avec des tableaux pour la mise en page
- Cela peut-être considéré comme un point fort : il embarque Apache, permettant de proposer un débogueur intégré, mais pas de se servir de son propre Apache.

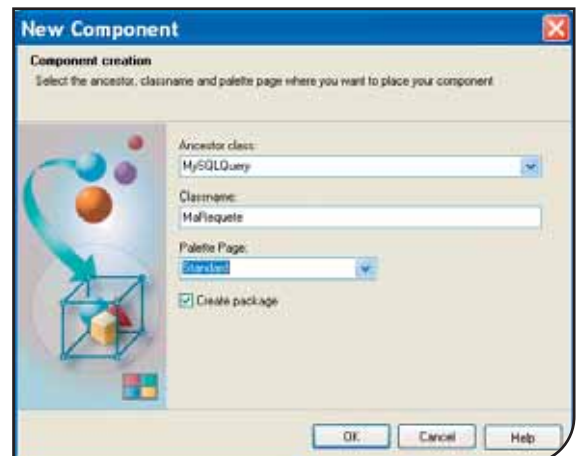
phpDesigner

Licence : commerciale (69 euros), personnelle (39 euros) ou étudiante (25 euros)
Editeur : MP Software (<http://www.mpssoftware.dk>)

L'outil tire son épingle du jeu, car il laisse beaucoup de configurations possibles pour le développeur, et propose une riche palette fonctionnelle !

Points forts :

- Il parse les erreurs et on peut le relier à la documentation au format .chm ou



- Tout est paramétrable, les préférences sont riches en personnalisations et explicites

Points faibles :

- disponible pour Windows uniquement
- il utilise l'OS pour afficher les fichiers dans le listing des fichiers du projet : on voit donc les fichiers avec le logiciel affecté par défaut.
- Trop de menus (cf copie d'écran 1) on est vite perdu
- Déclare une erreur alors que la complétion est visible et qu'on n'a pas fini de taper la ligne.

phpEdit

Licence : commerciale

Editeur : WaterProof (<http://www.phpedit.com/>)

Le plus célèbre et un des plus anciens éditeurs de code dédié à PHP. PhpEdit reste un outil à essayer car il est dédié exclusivement au PHP. Vaut-il le prix de sa licence face à des outils opensource ?

Points forts :

- choix de la langue possible
- peu gourmand en ressources
- mise en valeur contextuelle du code (pour les fichiers dans lesquels on mixe du HTML et du Javascript, par exemple)

Points faibles :

- nécessite le framework .Net 2.0 pour s'installer
- quelques plantages

EDI basés sur Eclipse

Eclipse est l'un des environnements de développement les plus connus. S'il est surtout connu pour le côté Java, il gère d'autres langages notamment web comme PHP, HTML, XML, javascript ou encore CSS.

Pour faire du PHP dans Eclipse, deux plugin open source existent : PHPEclipse et PDT. Notons que Zend Studio est aussi distribué sous forme de plugin. Tous ces outils permettent de profiter des fonctionnalités natives de l'environnement, à savoir :

la gestion par projets

- encodage choisi au niveau du projet
- import/export de projet depuis un projet Eclipse
- construction de projet depuis un répertoire pré-existant

et les fonctions intelligentes telles que :

- bookmarks dans le code,
 - web browser interne (une des vues) basé si besoin sur votre navigateur,
 - les marqueurs de tâche : on définit des mots-clefs (par défaut TODO et FIXME), qui vont déclencher un marqueur dans le code, puis être repris, selon leur priorité dans la vue Tasks
- Les quelques points faibles de ces outils sont :
- pas de version française
 - Eclipse crée un fichier .project dans votre répertoire

PHPEclipse : plugin PHP

Points forts :

- possibilité d'atteindre les éléments (fichiers inclus, déclaration de fonc-

tion, de classe, de constante, de variable, etc.) par un raccourci CTRL + clic : cela exécute en réalité une recherche incrémentale dans les fichiers du projet

- le code PHP est directement parsé, les erreurs de syntaxe sont détectées immédiatement et mises en valeur
- tous les points forts liés à Eclipse : possibilité de définir des Tags (TODO, DEBUG, ...)

Points faibles :

- la complétion n'est pas contextuelle quand on fait de l'objet : il propose donc toutes les variables du projet, toutes les méthodes de tous les objets du projet

PDT : PHP Development Tools plugin distribué par Zend

Points forts :

- explorateur de classes, et de code
- s'interface parfaitement avec Zend Framework et Zend Toolbar
- possibilité de définir des includes path propres au projet : cela permet de profiter des liens (par CTRL+clic) sur les classes parentes d'un framework.

Points faibles :

- debugger difficile à configurer

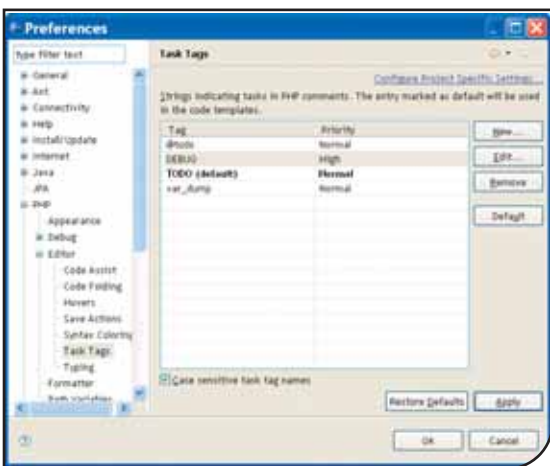
Conclusion

Utilisant habituellement des outils opensource, j'ai été agréablement surprise par le niveau de finition de certains logiciels, qui ne m'avaient pas impressionnée dans leurs premières versions.

Il existe assez d'outils pour que chacun y trouve son compte, et ce n'est qu'après une utilisation régulière qu'on s'approprie son EDI. Il vous faudra donc tester au moins deux outils pour trouver celui qui vous convient le mieux. Sachez qu'il ne faut pas hésiter à personnaliser les raccourcis, et jouer des préférences pour paramétrer au plus ergonomique son logiciel de développement.

■ Sarah Haïm-Lubczanski

Formatrice certifiée PHP / Consultante technologies web Open Source, Formatrice Ajax, web 2.0, Standards du web
sarah.haim@anaska.com



Monitorer et optimiser vos architectures Web

Une des problématiques les plus complexes d'un projet web de taille importante est la gestion de la montée en charge. En effet, il est difficile de prévoir quel aspect de l'architecture renforcer, ou bien quelle partie du code optimiser afin de gagner un maximum de performances. Au travers de cet article je vous propose de partager 12 ans d'expérience d'administration de serveurs Web.

Cycle de vie d'une application Web

En général on commence l'administration d'un serveur Web de façon très simple : on s'installe en local une version de développement (un serveur LAMP voire WAMP (Apache MySQL PHP)) puis on fait ses développements dessus et on envoie en production.

Les premières phases de vie d'une application ou d'un site peuvent généralement se faire sur un hébergement mutualisé (sauf besoins spécifiques) : c'est plus intéressant en termes de coût et de suivi. Les limitations d'un hébergement mutualisé touchent à la configuration PHP qui vous est proposée : elle est packagée et difficilement modulable.

Si votre application Web devient populaire vous allez vite vous sentir à l'étroit sur votre hébergement mutualisé (voire votre hébergeur vous fera comprendre qu'il faut passer au niveau supérieur !). Direction le serveur dédié ou le serveur virtuel. C'est là que les choses se corsent : il va falloir administrer et monitorer votre serveur, et débiter enfin les choses sérieuses.

Mesurer la performance

Pour faire vivre une application Web, mieux vaut prévenir que guérir et donc il faut analyser et monitorer votre architecture afin d'en trouver les points faibles. Il existe deux types d'outils pour cela :

- les outils de suivi des constantes
- les outils de mesure instantanée

Les outils de suivi des constantes

Disposer d'une valeur instantanée pour certaines constantes n'est pas suffisant. Il est important de connaître l'évolution de ces constantes sur une période plus ou moins longue, allant d'une journée glissante, à plusieurs semaines/mois. Pour cela, il existe différents outils : cacti/rrdtools ou MRTG pour ne citer que les plus connus. Les deux solutions permettent de générer des graphes sur toutes les constantes intéressantes sur une machine :

- trafic
- mémoire
- charge
- espace disque

Ces solutions sont basées sur le protocole SNMP. Donc, toutes les informations présentes au niveau SNMP peuvent être transformées en graphe. [Fig.1]

L'intérêt de ce suivi est de disposer d'une tendance à court, moyen et long terme. Cela permet d'anticiper une évolution de l'architecture, avant d'atteindre des problèmes de performance. Pour MySQL, cacti/rrdtools dispose de templates spécifiques pour la génération de graphes spéci-

fiques. On peut également citer l'outil *Mysql monitor*. Les graphes permettent une analyse fine des performances de la base.

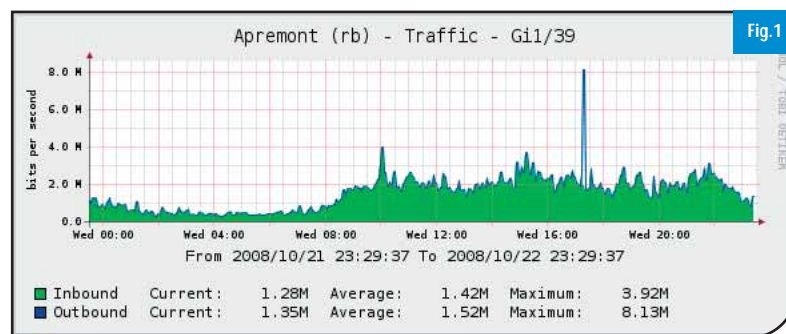
Les outils de mesure instantanée

Afin de cibler des problèmes de performance au niveau CPU, vous pouvez utiliser la commande Unix bien connue, 'top'. Elle permet d'obtenir des informations sur l'utilisation de chaque CPU, notamment avec l'option '1' (touche 1 lors de l'exécution de top).

Cette commande vous permettra de détecter quel processus consomme le plus de CPU, voire sature votre processeur. Il est intéressant notamment de regarder la valeur du CPU idle. Cette valeur correspond au pourcentage de temps où le processeur n'est pas occupé. Dans le cas d'une valeur nulle, cela indique une saturation complète du processeur.

Il existe une évolution de la commande top qui s'appelle *htop*. Elle est disponible pour toutes les distributions Linux. Bien qu'utilisée en console, *htop* est un peu plus 'graphique' que la commande top. [Fig.2]

Les commandes 'top' ou 'htop' per-



mettent également de connaître l'utilisation de la mémoire à un instant t. Elle permet donc d'identifier une saturation de cette dernière.

Pour la mesure instantanée des problématiques réseau, l'outil " iptraf " est indispensable. Il permet de connaître la valeur instantanée de la bande passante consommée sur chacune des interfaces réseau, *vlan* compris. Il permettra de mettre en avant toute saturation au moment de la consultation. [Fig.3]

Il existe également des outils de surveillance spécifique pour chaque application. Dans le monde MySQL, la commande 'mytop' permet de disposer d'informations synthétiques sur les requêtes en cours d'exécution et sur les différentes constantes de la base de données. Pour une utilisation plus spécifique avec les tables InnoDB, il existe également l'outil 'innotop'

Optimisations physiques

Avec les outils et techniques précédemment cités vous devriez avoir détecté les éventuels points de blocage, voyons comment résoudre ceux qui dépendent de l'architecture physique. Prenons le cas de la **saturation CPU**. C'est souvent cette problématique qui impose une évolution de l'architecture.

La première solution consiste à augmenter la puissance de la machine: augmentation de la fréquence processeur, doublement du nombre de processeurs. Mais il arrive un moment

ou l'on atteint la limite maximum de la puissance du matériel actuel.

Séparation et répartition des services

Une fois atteintes les limites liées aux améliorations physiques, il faut commencer à **séparer les services** sur différents serveurs. Dans le cas d'une architecture PHP/MySQL, cela consistera à délocaliser la base de données MySQL sur un serveur spécifique : MySQL peut alors tirer toute la puissance de la machine qui lui est dédiée. Si cette dernière opération est déjà faite, c'est-à-dire que chaque fonction de l'architecture dispose de son propre matériel, il faut alors répartir la charge sur plusieurs machines ayant la même fonction.

C'est là qu'il devient nécessaire de faire entrer en jeu un **répartiteur de charge**.

Optimisation matériel -> séparation des services -> répartition de la charge.

Le répartiteur de charge va permettre d'augmenter une nouvelle fois la puissance disponible pour chaque fonction de l'architecture, de manière totalement transparente pour la partie applicative.

Il existe plusieurs possibilités pour faire de la répartition de charge, des solutions propriétaires comme des solutions libres. Dans les solutions libres, nous pouvons citer 'ipvsadm' avec la surcouche 'keepalived' qui est très couramment utilisée.

Grâce à ipvsadm/keepalived, il est possible d'avoir plusieurs frontaux

web par exemple, et de répartir la charge selon certains critères sur les différents serveurs.

Il existe plusieurs algorithmes de répartitions de charge. Les plus connus sont :

- RoundRobin (ou Weighted RR): répartition équitable (ou proportionnellement au poids) des requêtes sur les serveurs disponibles
- LeastConnexion (ou Weighted LC): répartition des requêtes sur le serveur ayant le moins de connexions actives.
- ShortestExpectedDelay: Les nouvelles requêtes sont assignées à la machine qui aura le délai de réponse estimée le plus court.

Là où un serveur frontal pouvait accepter X requêtes entrantes, le fait d'utiliser une répartition de charge avec 3 frontaux vous permettra de disposer d'une puissance 3 fois supérieure. Mais renforcer une partie de l'architecture peut souvent provoquer des problèmes par ailleurs. Dans notre cas PHP/MySQL, un renforcement de la puissance des frontaux provoque une charge beaucoup plus importante sur la base de données.

Optimiser la base de données

Il est également possible d'intervenir à ce niveau là. Plusieurs solutions existent. La première d'entre elles consiste à utiliser une deuxième machine MySQL et dupliquer les données à l'aide de la réplication.

Il faut alors répartir les accès à la base. Cela demande une modification

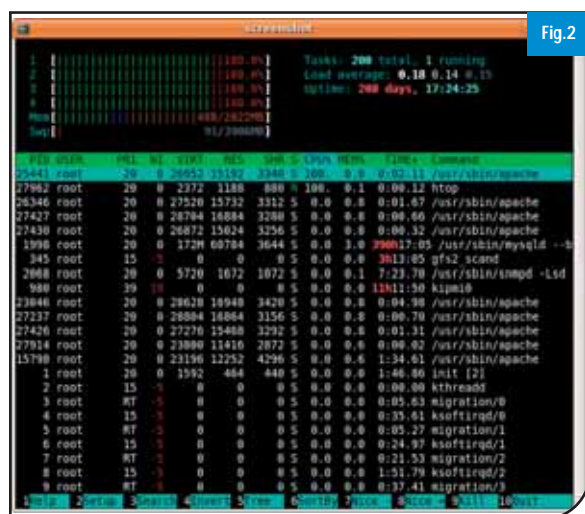


Fig.2

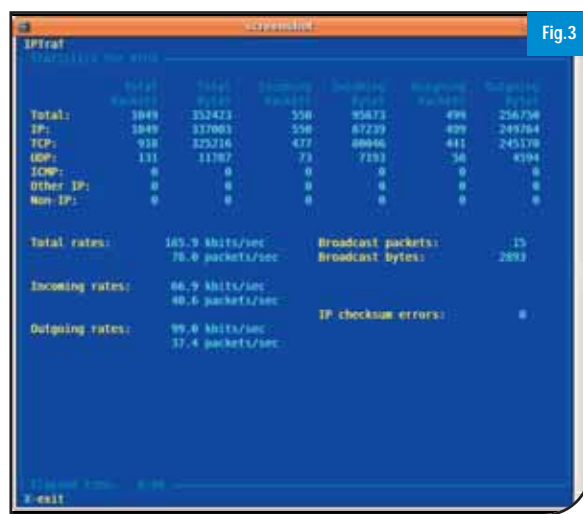
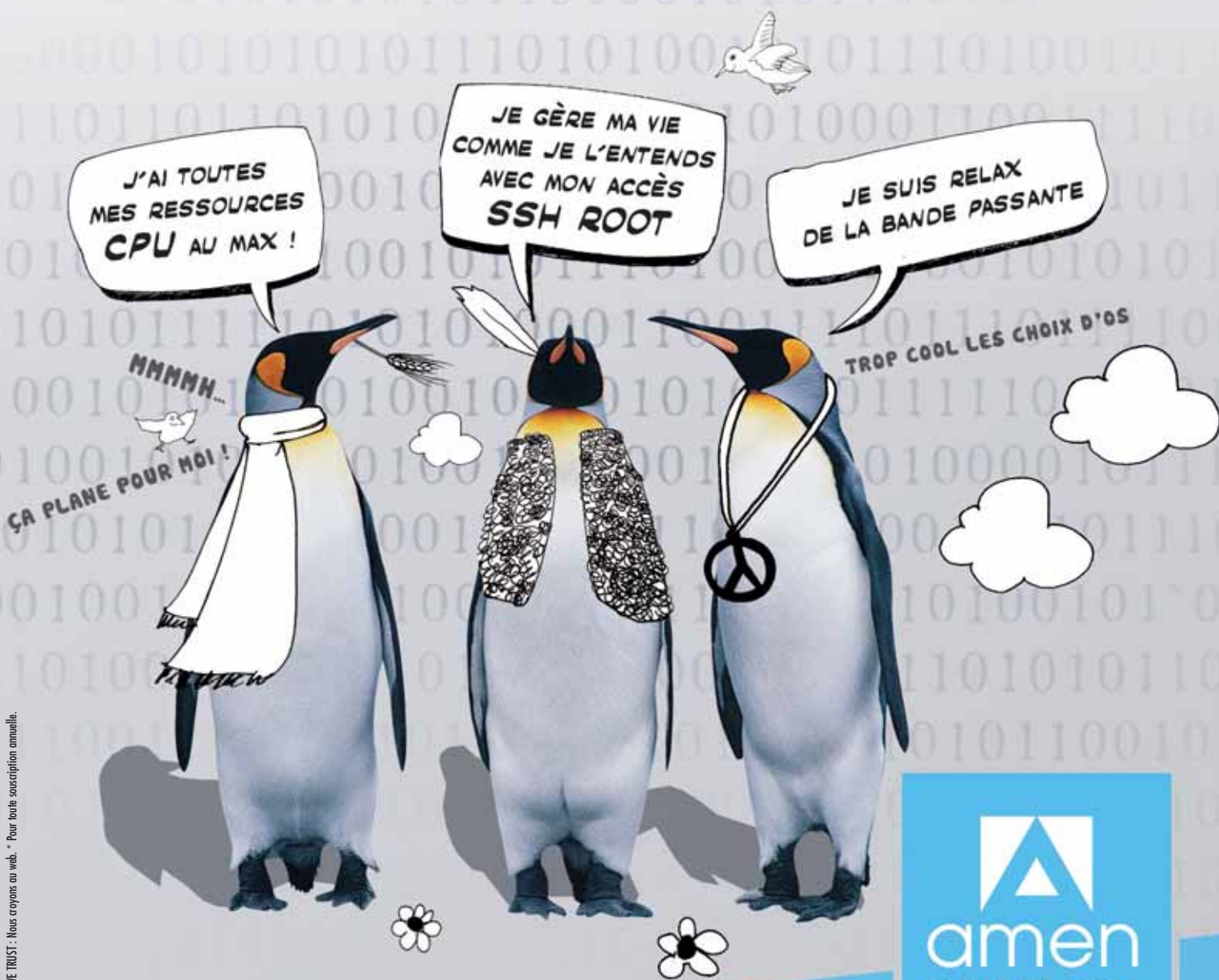


Fig.3

NOUVEAU VDS+ d'AMEN : le bonheur est dans le serveur !



À PARTIR DE
5€^{HT} /MOIS
soit 5,98 € TTC/MOIS*

**SERVEURS PRIVÉS AMEN :
BÉNÉFICIEZ DE RESSOURCES
GARANTIES QUI VOUS SONT
PROPRES (PROCESSEUR,
MÉMOIRE, DISQUE DUR...)
TOUT EN PROFITANT D'UNE
PLATEFORME INFOGÉRÉE
24H/24 - 7J/7.**

- Hébergement multi-sites/multi-domaines
- Interface d'administration : Plesk 8.6
- Systèmes d'exploitation : Fedora Core 8, Suse 10.3, Debian 4.0, Ubuntu 8.04 ou CentOS 5
- Part CPU minimum : de 1 à 6
- Mémoire garantie : de 256 Mo à 1 Go
- Espace disque : de 5 Go à 30 Go
- Bases de données : illimitées
- 1 adresse IP fixe
- Accès Root

**PARTENAIRE
INDUSTRIEL**



Pour plus de renseignements : 0892 55 66 77 (0.34 €/mn) ou www.amen.fr
NOMS DE DOMAINE - EMAIL - HÉBERGEMENT - CRÉATION DE SITE - E-COMMERCE - RÉFÉRENCIEMENT

applicative, afin de concentrer les requêtes de modifications (INSERT/UPDATE/DELETE/etc.) sur le serveur maître, et les requêtes de lecture (SELECT) sur le maître et sur l'esclave. Cette solution peut être particulièrement coûteuse en terme de développement, voire impossible pour des applicatifs trop anciens.

Une solution possible est d'utiliser un proxy d'accès à la base MySQL, comme mysqlproxy. Il permet, sans modifier l'application existante, à l'aide d'un langage spécifique, de répartir les requêtes MySQL entre plusieurs serveurs. Le seul point noir de cette solution est d'ajouter un point de failure supplémentaire au niveau de l'architecture, à moins de redonder le serveur mysqlproxy.

Optimiser l'architecture software

Intervenir au niveau matériel représente une possibilité d'optimisation. Une autre piste importante consiste à optimiser la puissance utilisée par votre application. Pour cela, il existe de nombreuses solutions de cache. Elles permettent de libérer de la puis-

sance, en ne refaisant pas certains calculs déjà réalisés auparavant.

Dans le monde PHP, il existe notamment le cache *d'opcode* qui permet cela. Lors de la première demande d'une page, le système va transformer en *opcode* le code PHP. Cet *opcode* sera stocké en mémoire pour une réutilisation future. Ce type de solution permet d'avoir des gains impressionnants sur des applications complexes.

Il existe plusieurs développements libres offrant cette fonctionnalité. Pour gérer nos hébergements chez NexenServices, nous utilisons essentiellement les deux suivants :

- xcache (<http://xcache.lighttpd.net/>)
- apc (<http://fr.php.net/apc>)

Si les gains sont très intéressants d'un point de vue sauvegarde des ressources, il peut apparaître, dans le cas de code PHP très complexe, des instabilités particulièrement gênantes et difficiles à reproduire.

Toutes ces solutions d'optimisation / gains de performance tentent d'arriver à la solution ultime qui consiste à servir des pages HTML. En effet, c'est avec ce type de solution que l'expérience utilisateur sera la meilleure, car l'affichage sera le plus fluide possible. Lorsque cela est possible, générer les pages de votre site en HTML et servir ses pages HTML aux internautes est la meilleure solution possible. C'est souvent les solutions qui sont adoptées par les différents Framework / CMS.

Optimisation système

Au niveau système, une optimisation qui est de plus en plus fréquente consiste à demander à Apache de traiter les aspects de génération des pages du site uniquement. Tout ce qui

concerne les médias est alors géré par un serveur web différent d'Apache, qui sera plus optimisé pour ce type de traitement. Dans cette catégorie de serveur web, on trouve notamment *lighttpd*.

L'intérêt de ce type de solution est que le serveur web qui traite les images a une empreinte mémoire très faible. Il n'embarque que des fonctionnalités réduites, et surtout ne charge pas de module permettant de gérer le PHP. Au contraire, Apache qui va gérer les pages PHP ne sera pas occupé une partie du temps à gérer les fichiers médias (images/javascripts/css/flash).

La mise en place de ce type de solution demande une configuration particulière pour le développement du site. En effet, il faut rassembler tous les médias sur un sous-domaine particulier. Il est quand même possible de rester sous le sous-domaine principal et de répartir la charge entre Apache et lighttpd suivant les répertoires. Cela demande la mise en place d'une répartition / distinction du flux au niveau de la couche 7 OSI. Cela peut s'avérer assez lourd.

Conclusion

La montée en charge d'une application web doit être anticipée au maximum, afin que l'évolution de l'architecture ne soit pas un point pénalisant. Prendre des bonnes directions dans le développement dès le départ revêt donc un caractère très important.



■ Ghislain Seguy
Directeur technique de la société Nexen Services du groupe Alterway, spécialisé dans l'hébergement à valeur ajoutée.

Choisir son hébergeur Quels critères ?

Quels sont les critères pour choisir un bon hébergeur ? Tout dépend de ce que vous voulez et de l'importance de l'application. En général on pense d'abord au prix mais il faut se poser la question du coût d'une indisponibilité. Si quelques heures d'indisponibilité ne vous gênent pas, alors la solution du mutualisé peu cher est certainement la plus logique. Par contre, si vous jugez que ne pas avoir accès à vos e-mails pendant une demi-journée est critique, alors optez pour une solution proposant un support garanti.

- disponibilité,
- prix,
- outils et configuration,
- support.

NOUVEAU !
TRIBUNE LIBRE
www.programmez.com

NOUVEAU !
L'ACTUALITÉ en VIDÉO
www.programmez.com

Manipuler des fichiers OpenDocument

OpenDocument est le format de fichier novateur utilisé par OpenOffice. Outre le fait qu'il soit un standard ISO des fichiers bureautique, il a pour point fort d'être en XML ! Qui dit XML dit lisibilité et opérabilité simplifiée ! Dans cet article nous allons mettre en avant une solution écrite en PHP qui a profité de l'ouverture de ce standard pour créer une bibliothèque permettant de générer des fichiers OpenDocument.

Pour comprendre comment interagir avec des fichiers au format OpenDocument il est nécessaire de comprendre comment est architecturée cette norme. Passons donc aux choses sérieuses, analysons un fichier bureautique XML : un fichier traitement de texte d'OpenOffice (extension odt). [Fig.1]

Un fichier odt est en fait un fichier compressé comprenant plusieurs fichiers et répertoires. Si vous disposez d'OpenOffice faites le test vous même : créez un fichier simple avec OpenOffice, enregistrez le, fermez OpenOffice et dézippez le fichier (utilisez le logiciel Open Source 7zip si vous n'avez pas de logiciel installé). La figure 1 vous montre la structure d'un fichier odt.

Dans le fichier "content.xml" on retrouve tout le contenu textuel du fichier. C'est sur ce fichier que l'on peut agir pour faire des modifications textuelles. Le répertoire "Pictures" contient les images de votre fichier. Pour l'instant nous nous contenterons de ces informations et allons voir comment une bibliothèque PHP relativement simple peut nous permettre de générer des fichiers OpenOffice sur un modèle de templating.

Présentation de la bibliothèque odtPHP

OdtPHP est une bibliothèque PHP destinée à servir de moteur de template pour des fichiers OpenDocument. Son fonctionnement est similaire à du templating classique : vous créez un modèle de fichier Writer avec OpenOffice et vous insérez dedans des balises spéciales aux endroits où vous souhaitez mettre du contenu spécifique.

Ce fichier servira de "modèle" pour la génération de fichiers "finis" via la bibliothèque odtPHP. Son utilisation requiert PHP 5.2 et d'activer l'extension Zip.

Notre tutoriel se base sur la version 0.2 qui supporte :

- la gestion des remplacements simples
- la gestion des boucles
- la gestion des images [Fig.2]

Site de téléchargement : <http://sourceforge.net/projects/odt-php/> ou sur notre CD-Rom ou notre site.

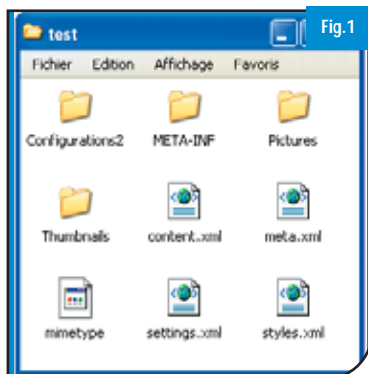
Commencez par télécharger le fichier contenant la bibliothèque, dézippez le fichier et vous devriez avoir trois répertoires comme le montre la [Fig.3].

Dans le répertoire "tests" vous avez trois tutoriels qui vous permettront de mettre en pratique les trois principales fonctionnalités de base de odtPHP.

Pour utiliser la bibliothèque il vous suffit de copier l'ensemble du répertoire "library" dans votre arborescence Web (ou, pour plus de sécurité, hors de l'arborescence Web, dans un répertoire contenu dans la directive de configuration "include_path" du php.ini).

Mise en place du template OpenOffice

La première phase consiste à mettre en place votre document référent OpenOffice. De la même façon qu'avec les templates PHPLib il vous suffit d'indiquer les endroits que vous souhaitez remplacer avec la notation suivante "{titre}", "{message}", ... La figure 4 vous montre un exemple simple de fichier OpenOffice avec des tags de type template. Enregistrez ce fichier dans le répertoire ou



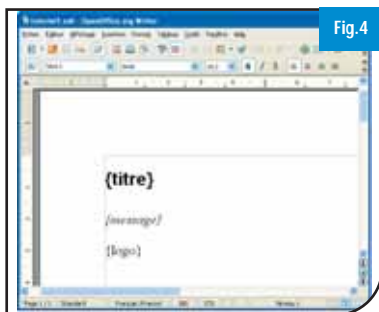
Structure d'un fichier OpenDocument



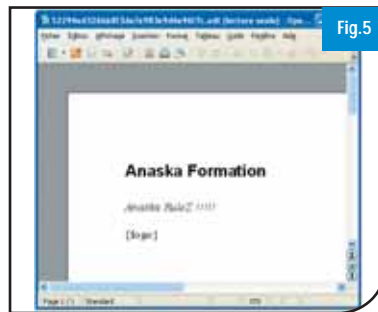
OdtPHP sur sourceforge



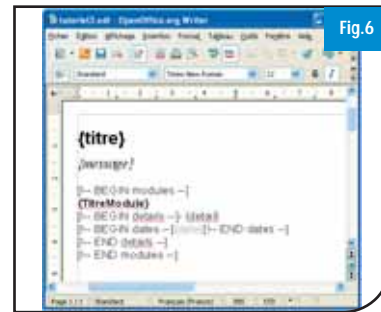
Contenu du dossier odtPHP récupéré sur sourceForge



Exemple de fichier OpenOffice simple



Exemple de rendu OpenOffice



Exemple de template OpenOffice

se trouve votre bibliothèque odtPHP, notamment le fichier "odf.php"

Appels à la bibliothèque

script PHP pour un document OpenOffice avec remplacement simple

```
<?php
// First : include the library
require_once('..odf.php');
$odf = new odf("tutoriel1.odt");
$odf->setVars('titre', 'Anaska Formation');
$odf->setVars('message', "Anaska RuleZ !!!!");
// On exporte le fichier
$odf->exportAsAttachedFile();
?>
```

Ces quelques lignes de code vont vous permettre d'utiliser la fonctionnalité la plus basique de la bibliothèque odtPHP : le remplacement de variables. La figure 5 vous montre le rendu associé à ce bout de code.

Si vous vous demandez comment odtPHP gère cela, rien de plus simple une expression régulière recherche notre variable et la remplace dans le fichier content.xml.

Mettre en place une boucle

Maintenant que nous savons comment faire un remplacement simple, voyons comment nous pouvons mettre en place plusieurs lignes de texte. Pour commencer, il faut indiquer dans notre fichier OpenOffice ou nous souhaitons avoir des boucles et le modèle des lignes que nous souhaitons mettre dedans. La ligne "[[BEGIN dates -]] Prochaine date : {date} [[END dates -]]" permettra d'indiquer que nous souhaitons voir apparaître une liste de dates suivant le modèle défini. La figure 6 nous montre un exemple avec deux boucles imbriquées. Le script PHP pour remplacer les templates dans le document OpenOffice est le suivant :

```
<?php
require_once('..odf.php');
$odf = new odf("tutoriel3.odt");

$odf->setVars('titre', 'What about trying with more contents ?');
$odf->setVars('message', "Hum it s cool, \nOpen Source will rule the world.");
```

```
$j=1;
$module = $odf->setSegment('modules');
while ($j < 3) {
    $i = 1;
    $module->setVar('TitreModule', 'Titre de module' . $j);
    while ($i < 4) {
        $module->details->detail('Anaska is the best Open
Source ! ' . $i);
        $module->details->dates->date(date('d/m/Y'));
        $i ++;
        $module->details->merge();
    }
    $j ++;
    $module->merge();
}
$odf->mergeSegment($module);
// On exporte le fichier
$odf->exportAsAttachedFile();
?>
```

Aller plus loin

Le dernier répertoire que vous récupérez sur sourceforge est le répertoire " documentation ". Tout le code de la bibliothèque a été écrit au format PHPDoc qui permet de générer facilement de la documentation.

Limitations de la bibliothèque

La bibliothèque odtPHP a été développée en France par les équipes d'Anaska Formation pour répondre à certains de leurs besoins spécifiques. Le code a été mis à disposition pour la communauté mais il peut ne pas marcher dans le cas de remplacement dans des fichiers complexes. La version 0.2 fonctionne sur le principe d'expressions régulières qui peuvent encore être améliorées. Une première tentative avait été faite via une manipulation de l'arbre XML en DOM, mais l'implémentation avait été jugée trop lourde par rapport au besoin. Comme dans tous projets OpenSource, toute aide est bienvenue.



■ Cyril Pierre de Geyer est responsable technique d'Anaska, spécialisée dans la formation aux technologies Open Source (Groupe Alter Way). Cofondateur de l'AFUP, il participe aux sites PHPteam.net et PHPFrance.

Mon premier projet web avec Symfony

Symfony est un framework de développement web programmé en PHP5. Un framework est le montage d'un ensemble d'outils de développement permettant d'éviter aux programmeurs les tâches répétitives sur lesquelles ils n'ont que très peu de valeur ajoutée.

Nous verrons comment Symfony nous permet de programmer une application propre et maintenable en un minimum de temps. Je vous propose de réaliser une petite application web permettant à des internautes de s'inscrire à des sessions de stage. Nous verrons étape par étape les phases d'installation et de développement.

1 PRÉ-REQUIS SYMFONY

Il existe plusieurs moyens de l'installer sur un ordinateur et de le coupler avec des *projets*. Le moyen le plus simple de débiter est d'installer une *sandbox*, c'est un environnement prêt à l'emploi comprenant un projet et une *bibliothèque* Symfony pré-configurés et prêts à l'emploi. Quels sont les pré-requis pour utiliser la *sandbox* Symfony ?

- Un serveur web capable d'exécuter des scripts PHP (>=5.2.x)
- Le module PHP d'accès aux bases sqlite (base utilisée par la *sandbox*)
- L'accès à PHP en ligne de commande (de préférence dans le Path)
- Le module *rewrite* d'Apache (optionnel mais recommandé).

Il est possible de télécharger la *sandbox* Symfony à l'adresse : http://www.symfony-project.org/get/sf_sandbox_1_1.tgz

2 INSTALLER NOTRE PROJET

Il suffit de décompresser l'archive de la *sandbox* dans le répertoire racine de votre serveur web. Cela devrait ressembler à cela : [Fig.1] Ne vous laissez pas intimider par le grand nombre de répertoires et/ou de fichiers, cela signifie tout simplement que tout a une place bien précise qui est toujours la même quels que soient les projets. Le répertoire *apps* contient les applications de notre projet. Par défaut, la *sandbox* est livrée avec une application nommée *frontend*, il y a de grandes chances qu'une application *backend* y trouve sa place à un moment où à un autre. Je pense que vous n'aurez aucun mal à deviner la fonction des répertoires *data*,

cache, *config*, *doc*, *lib*, *plugins* et *test*. Le répertoire web est dédié à être le répertoire racine d'Apache. Cela permet de garder l'ensemble des fichiers du projet hors de la portée des internautes, ce qui est toujours une garantie de sécurité intéressante. Un fichier *.htaccess* s'y trouve qui réécrit les URLs pour demander à Apache de charger le fichier *index.php* quoiqu'il arrive, symfony s'occupera de savoir quoi faire avec l'URL. Fixer le *documentRoot* d'Apache au sous-répertoire *web* n'est pas obligatoire bien que ce soit fortement recommandé.

Si tout est configuré correctement, vous devriez voir l'écran suivant dans votre navigateur : [Fig.2]. Il est possible que les utilisateurs Unix (Linux et OS X) aient une exception à cause des droits d'écriture d'Apache dans le répertoire de cache qui ressemble un peu à cela : " sfCacheException: Failed to make cache directory "/Un/répertoire/de/votre/disque" while generating cache for configuration file "config/config_handlers.yml". In ... " Ne paniquez pas (encore), nous allons voir juste en dessous comment tout remettre en ordre, le temps de vous le présenter.

3 L'OUTIL DE LIGNE DE COMMANDE

À la racine du projet, vous trouverez un fichier nommé *symfony*. Cet outil permet d'appeler les automatismes de Symfony en ligne de commande. Il suffit, dans un terminal, de se placer dans le répertoire du projet et de l'interpréter avec PHP pour s'en servir :

```
$ php symfony --version # pour toutes les plates-formes
symfony version 1.1.4 (/var/www/sf_sandbox/lib/symfony)
$ ./symfony --version # pour Unix, Linux, OSX etc.
symfony version 1.1.4 (/var/www/sf_sandbox/lib/symfony)
```

Les personnes ayant eu droit à l'exception à cause des permissions peuvent résoudre leur problème en tapant la ligne suivante :

```
$ ./symfony project:permissions
```

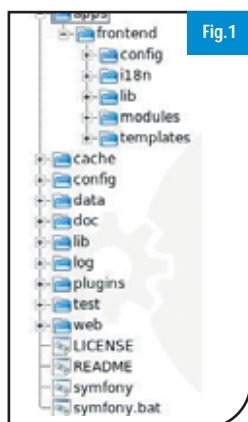


Fig.1



Fig.2

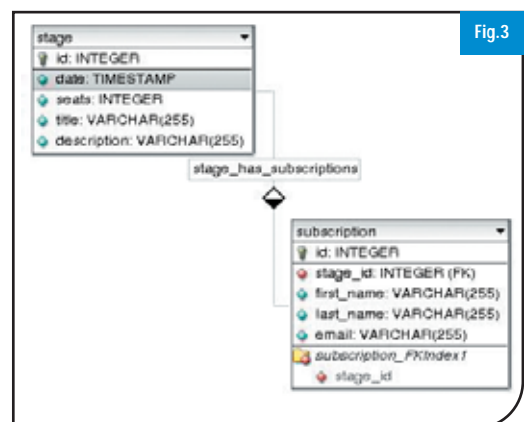


Fig.3

Cet outil en ligne de commandes sera très utile tout au long du développement de notre application, nous l'utiliserons plusieurs fois au cours de ce développement.

4 LE SCHÉMA DE BASE DE DONNÉES

Le schéma de base de données se définit dans le fichier `/config/schema.yml`. La très grande majorité des fichiers de configuration de Symfony sont au format YAML (<http://fr.wikipedia.org/wiki/Yaml>). Attention donc à l'indentation (2 espaces) et à ne pas y insérer de TABulations. Nous allons définir le schéma suivant : [Fig.3]

```
propel:
  stage:
    id:
    date: timestamp
    seats: integer
    title: { type: varchar(255), required: true }
    description: longvarchar
  subscription:
    id:
    stage_id:
    first_name: { type: varchar(255), required: true }
    last_name: { type: varchar(255), required: true }
    email: { type: varchar(255), required: false }
```

Comme vous le voyez, nous laissons Symfony deviner le type et la fonction de certains champs. Il reconnaît la clé étrangère de l'inscription vers la table stage grâce à son nom. Pour demander à Symfony de construire les classes du modèle il suffit d'utiliser l'outil en ligne de commande :

```
$ ./symfony propel:build-all
```

Il se peut que quelques messages d'erreurs s'affichent la première fois lorsque le gestionnaire de la base de données cherchera à détruire les tables pour les reconstruire, vu qu'elles n'ont jamais encore été créées. Relancez la commande, les erreurs ne devraient pas ré-apparaître. [Fig.4]

5 CRÉER DES DONNÉES DE TEST

Les données de test permettent d'alimenter automatiquement notre application à chaque fois que nous changerons le modèle ou chercherons à tester notre application. Pour cela, il est nécessaire de créer le répertoire `/data/fixtures` et d'y placer un fichier au format Yaml contenant les données voulues.

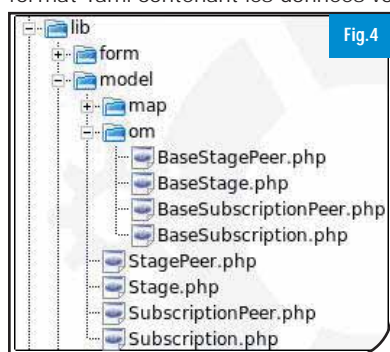
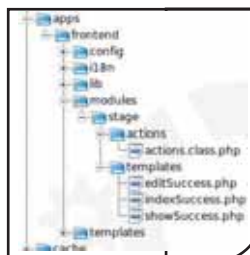


Fig.4



```
Stage:
  stage1:
    date: 2008/12/01
    seats: 10
    title: super stage
    description: Ceci est un super stage
  stage2:
    date: 2008/12/15
    seats: 10
    title: stage ennuyeux
    description: Ceci est stage ennuyeux
Subscription:
  subscription1:
    stage_id: stage1
    first_name: alpha
    last_name: one
    email: alpha.one@monmail.com
  subscription2:
    stage_id: stage1
    first_name: bravo
    last_name: two
    email: bravo.two@monmail.com
/data/fixtures/init.yml
```

Pour insérer ces données, la ligne de commande symfony va encore une fois faire le travail à notre place. À noter que Symfony utilise dès lors le modèle précédemment généré. Il peut s'avérer nécessaire de réinitialiser le gestionnaire d'auto-loading des classes en effaçant le cache :

```
./symfony cc # efface le cache
./symfony propel:data-load frontend # insère les données
```

6 GÉNÉRER LE GESTIONNAIRE DE STAGES

Symfony propose un certain nombre de générateurs de code. Parmi ceux-ci, la génération de CRUD qui permet de construire automatiquement les mécanismes permettant les *Create*, *Retrieve*, *Update* et *Delete* sur un objet précis, à savoir ici un objet *Stage*.

```
./symfony propel:generate-crud --with-show frontend stage Stage
```

(attention à la majuscule sur le dernier paramètre: "Stage"). Cela va vous créer un nouveau module nommé "stage" dans votre application frontend, ce que nous pouvons vérifier dans un explorateur de fichier et un explorateur web : [Fig.5]

Stage List

Id	Date	Seats	Title	Description
1	2008-12-01 00:00:00	10	super stage	Ceci est un super stage
2	2008-12-15 00:00:00	10	stage ennuyeux	Ceci est stage ennuyeux

Fig.5

file:///var/www/sf_sandbox

http://localhost/frontend_dev.php/post/index

Le résultat est un peu "brut de décoffrage". C'est exactement ce pourquoi le CRUD est fait : générer une base de travail sûre et malléable.

7 ENLEVER LE CODE "EN TROP"

Cependant, nous voulons seulement présenter une liste de stages et non pas permettre aux internautes de créer ou modifier des stages. Ceux-ci ne pourront que choisir un stage et s'y inscrire. Nous allons donc enlever le code de création / modification / suppression des stages dans le fichier `actions.class.php`.

Seules les méthodes `executeIndex` (qui affiche la liste des stages) et `executeShow` (qui affiche les détails d'un stage) nous sont nécessaires.

Par conséquent, nous n'avons également pas besoin de la template `editSuccess.php` que nous pouvons effacer. Dans les deux templates restantes, il ne reste qu'à enlever les liens (`link_to`) vers les actions que nous avons enlevées (`post/edit` et `post/create`). Le code de l'action devient alors :

```
class stageActions extends sfActions
{
    public function executeIndex()
    {
        $this->stageList = StagePeer::doSelect(new Criteria());
    }

    public function executeShow($request)
    {
        $this->stage = StagePeer::retrieveByPk($request->getParameter('id'));
        $this->forward404Unless($this->stage);
    }
}

/appsf/frontend/modules/stage/actions/actions.class.php
```

Nous n'avons pas encore écrit une ligne de code, nous en avons même enlevé.

8 AFFICHER LE NOMBRE DE PLACES RESTANTES

Sur notre page d'affichage des stages `indexSuccess.php` nous aimerions pouvoir afficher par exemple le nombre de places restantes. Nous allons pour cela nous reposer entièrement sur le modèle généré par Propel, le gestionnaire de modèle de Symfony. Si nous regardons la classe `BaseStage` dans `/lib/model/om` nous constaterons que Propel a bien identifié la clé étrangère vers les inscriptions et nous propose la méthode suivante : `countSubscriptions()`. Cela retourne le nombre d'inscriptions mais pas le nombre de places restantes associées à un stage. Nous allons donc écrire notre première ligne de code en enrichissant le modèle avec une nouvelle méthode que nous rajouterons dans la classe `Stage` qui hérite de `BaseStage` mais qui aura le mérite de ne pas écraser, si nous re-générons le modèle plus tard.

```
class Stage extends BaseStage
{

```

```
public function getAvailableSeats()
{
    return $this->getSeats() - $this->countSubscriptions();
}

/lib/model/Stage.php
```

Une fois armé de notre modèle, il ne reste qu'à changer dans la template l'utilisation de `getSeats()` par `getAvailableSeats()`.

9 AFFICHER LE DÉTAIL DES INSCRIPTIONS

Sur la page d'affichage d'un stage, nous désirons afficher la liste des inscrits. La question ici est que nous voulons afficher des inscriptions. On peut en effet en PHP5 définir la représentation d'un objet en chaîne de caractères à l'aide de la méthode spéciale `__toString()`. Nous pouvons choisir de retourner une chaîne du type "Nom Prénom".

```
class Subscription extends BaseSubscription
{
    public function __toString()
    {
        return ucwords($this->getFirstName(). ' '.$this->getLastName());
    }
}

/lib/model/Subscription.php
```

De la même façon que dans l'étape précédente, nous remarquons que *Propel* a généré pour nous une méthode `getSubscriptions()` dans la classe `Stage`. Nous pouvons l'utiliser directement dans notre template `showSuccess` en profitant de notre méthode `toString()` pour afficher directement les objets. À la fin de notre template, nous ajoutons alors l'itération sur les inscriptions.

```
Subscription List :
<ul>
<?php foreach ($stage->getSubscriptions() as $subscription): ?>
<li><?php echo $subscription ?></li>
<?php endforeach ?>
</ul>

/appsf/frontend/modules/stage/templates/showSuccess.php
```

CONCLUSION

Dans cet article, nous avons pu aborder les bases de Symfony, nous verrons dans un prochain article comment utiliser le framework de formulaires de Symfony 1.1 afin de laisser les internautes s'inscrire à nos stages.



■ Grégoire HUBERT

Consultant chez SensioLabs et mainteneur de la branche 1.0 du framework.

Windows 7, Windows Azure, Cloud Comp

La nouvelle Microsoft

Nous vous avons promis un mois de novembre .Net, promesse tenue ! Et Microsoft nous a bien aidés dans cette démarche avec de très nombreuses annonces autour des systèmes, des technologies, des langages, des outils.

Source image : World Wide Telescope

Si rien n'est exploitable, car tout est en CTP, en pré-bêta ou non encore disponible, 2009 promet beaucoup de bouleversements pour le développeur Windows, .Net, tout en "draguant" du côté des développeurs Java, Php, Ruby, Python... Une sacrée ouverture qui s'accélère depuis plusieurs semaines comme on a pu le constater avec le projet Eclipse dédié à Silverlight 2.

Première constatation: il existe une intégration profonde entre les modèles desktop, serveur, mobile et web. Désormais, la plate-forme Windows / .Net veut offrir une plate-forme de bout en bout avec un modèle de développement, un modèle applicatif, le plus unifié possible. Ainsi, une application desktop devra pouvoir fonctionner indifféremment sur le *nuage informatique* avec

Windows Azure même si là aussi, on bénéficie de SDK, de bibliothèques spécifiques car le contexte d'exécution n'est pas identique. Il s'agit aussi de sortir du navigateur les technologies, ainsi une application Silverlight 2.0 pourra fonctionner directement sur le desktop ! Jusqu'à présent, on ne voyait pas forcément où Microsoft allait avec les services en ligne, le modèle SaaS, les fonctions Live.

Aujourd'hui, nous le savons et de quelle manière ! Car avec Windows Azure, nous disposerons d'une plate-forme système dans le *nuage*, permettant d'exécuter nos applications. Et l'union de tous ces services, de .net et des outils qui permettront demain d'unifier les différents modèles de développement tout en simplifiant au maximum le codage. Bien entendu nous n'oublions pas

outing, Visual Studio 2010, Live, multicore

galaxie



Windows 7 qui modifiera aussi le comportement des développeurs.

A cela s'ajoutent trois autres défis majeurs : la modélisation, le parallélisme et VB. Sur la modélisation, OSLO devient la pierre angulaire de l'offre Microsoft. OSLO propose son propre langage de modélisation, le langage M. Il s'agit d'outiller, enfin, les Domain Specific Language (DSL). Outillage qui faisait défaut pour simplifier leur création. Nous ne sommes pas dans de l'UML qui reste pour le moment cantonné à Rosario, le futur Team System. Cependant, OSLO représente une plateforme centrale car toute la modélisation : système, infrastructure, déploiement, DSL, projet, passera par OSLO ! Côté parallélisation, Visual Studio 2010 ira bien plus loin avec de nouveaux mécanismes pour simplifier

le développement. Et les projets de type PLinq ou Parallel FX prendront toute leur place. Enfin, la PDC a pu répondre à une peur chronique : la mort de VB.Net. Microsoft a voulu être rassurant. Non, VB reste en vie et évoluera désormais au même niveau que C#, les deux équipes travaillant étroitement ensemble !

Voilà en quelques mots les annonces faites à la PDC et démontrées durant les TechEd de Barcelone. Dans ce dossier spécial, nous allons approfondir tout cela, et bien plus encore. Accrochez-vous car Microsoft promet réellement de vous envoyer dans les étoiles des galaxies *Cloud Computing* et .Net...

■ François Tonic

Microsoft dans les nuages !

Sur le CD
du numéro !



La Microsoft PDC (Professional Developers Conference) 2008 s'est déroulée du 27 au 31 Octobre dans le "Convention Center" de Los Angeles. Durant 5 jours, l'éditeur a dévoilé les prochaines technologies que nous allons utiliser. On peut retenir comme mots clés : Azure, services en ligne, .Net 4, modélisation, Windows 7.



La PDC est généralement organisée tous les 2 ou 3 ans et donne la ligne directrice de Microsoft sur les outils, les technologies et leur stratégie. Contrairement aux conférences Mix et TechEd, la PDC est avant tout un laboratoire de recherche sur le futur. Généralement

nous recevons des informations sur un scope temporel de 3 à 5 ans. La dernière édition datait de 2005 et se limitait à une vision sur deux ans alors que celle de 2003 portait sur une période de 5 ans avec des technologies comme Longhorn (Vista) ou encore Whidbey (Visual Studio)...

L'ambiance était chauffée par l'attente des annonces. Initialement partis pour découvrir en primeur Windows 7 ou encore Visual Studio 2010, le .NET Framework 4.0, voire les nouveautés des langages C# 4.0 ou Visual Basic 10.0, nous avons été plus que surpris par la société de Redmond ! Et les milliers de développeurs furent aux anges ! Durant la keynote inaugurale, le souffle et la fraîcheur de la vision de Ray Ozzie étaient plus que présents. L'ère Bill Gates est belle et bien terminée. Ceux qui doutaient de la capacité de Ray Ozzie à reprendre le flambeau et à jouer le rôle de Capitaine, afin de réinventer ce grand paquebot, peuvent être rassurés. Reste à transformer l'essai et à ne pas commettre les mêmes erreurs qu'avec Windows Vista.

Windows Azure : contre VMware et aller au-delà

Un des gros focus de la PDC et qui monopolisera une partie des regards concerne le nouveau Windows en ligne : Windows Azure. Pour faire simple, Azure est un système d'exploitation disponible directement en ligne, dans le *nuage* (ou *cloud compu-*

ting) pour reprendre le terme de Microsoft. Des rumeurs depuis septembre circulaient sur un tel système. Un peu comme VMware vCloud, Azure est une plate-forme composée, et proposant donc, des services pour le développeur, l'administrateur, l'utilisateur. Il se couple à une notion d'informatique à la demande. C'est-à-dire que l'infrastructure virtuelle monte en charge selon les besoins, pour le stockage, la puissance de traitement. Et tout cela, sans casser le modèle applicatif actuel : les développeurs utiliseront toujours Visual Studio (2008 actuellement, 2010 demain), .Net et les standards / protocoles du web tels que SOAP, REST, XML, etc. Sur tout, Microsoft a prévu dès l'origine l'interopérabilité et l'ouverture vers d'autres langages !

Azure représente une avancée majeure de la vision et stratégie Microsoft dans le monde du Software + Services et permet de mieux comprendre le discours évolutif de SOA vers SaaS (Software as a Service) et enfin vers S + S (Software + Services), tenu par la société Microsoft depuis quelques années déjà ! Comme quoi la stratégie était déjà bien présente de longue date et leur volonté, plus particulièrement celle de Ray Ozzie, était de réussir à surprendre la planète développement en sortant un panel évolué et complet d'outils du "On the Cloud". Effectivement cette extension apporte, au travers de Windows et de ses outils de développements pour le Cloud, la possibilité pour tout développeur de bâtir aisément des applica-



tions et services s'exécutant dans le Cloud.

Azure ne sera pas disponible en boîte comme Vista. Il le sera uniquement en ligne (via www.azure.com, avec un compte utilisateur). Pour le moment, seul Microsoft l'héberge et fournit les centres de données sur lesquels il fonctionne. Nous ne savons pas s'il pourra être déployé par d'autres prestataires que Microsoft.

Au-dessus d'Azure, nous disposons de toute une série de services (regroupés dans Azure Service Platform), à savoir : Live Services, .Net Services, SQL Services, SharePoint Services et Dynamics CRM Services. Dès aujourd'hui, les développeurs desktop et web disposent des SDK nécessaires pour développer ces premiers services !

L'ouverture d'Azure permet d'imaginer faire tourner dessus des applications, des sites PHP, du Java, du Python, du Ruby, etc. ! Et on peut même rêver disposer de templates projets aussi bien pour Visual Studio qu'Eclipse ! Le développeur .net reste cependant le focus principal. Et ses codes ASP.Net et de tout langage .Net fonctionneront dans Azure.

La particularité de ce système est de mettre à disposition :

- Stockage et hébergement à la demande
- Une forte capacité de montée en charge
- La possibilité de manager toutes applications dites online ou on the cloud.

Live Services

Les Live Services, mieux connus sous le nom de "Live Platform Services", sont un ensemble de "building blocks" permettant de lier les données utilisateur ainsi que les ressources applicatives dans la nouvelle plate-forme Azure. Leur objectif étant principalement de permettre la construction d'applications qualifiées de "réseau social" et de permettre de toucher la plus large audience Web possible.

.NET Services

Les .NET Services, au départ nommés "Biztalk Services" ou encore "Zurick", ont pour objectif de faciliter le développement des applications basées sur le "On the Cloud". Ainsi

les .NET Services intègrent en leur sein une série de fonctionnalités et couches fonctionnelles, comme l'accès sécurisé aux applications, un Service Bus afin de faciliter la communication inter applications / services ou encore le hosting et l'exécution de Workflow. Cette partie .NET Services permettra également à tout développeur de créer aisément des applications exploitant le modèle actuel et le modèle "on the cloud".

SQL Services

Les SQL Services, précédemment nommés "SSDS", ont pour objectif d'étendre les capacités de Microsoft SQL Server vers le "On the Cloud". Cette mise à disposition de la puissance de SQL Server au service des applications web permettra de travailler plus naturellement sur des notions de données relationnelles distribuées. Ce service met à disposition un Web Service permettant la recherche, les requêtes relationnelles, la synchronisation de données avec tous types de terminaux et/ou partenaires (Mobile, Remote Office, ...). Ce service permet enfin de retrouver n'importe quel type de jeu de données : structurées, semi-structurées ou encore non structurées.

Microsoft Sharepoint Services / Dynamics CRM Services

Dans le futur, les développeurs sur la plate-forme Azure pourront accéder aux fonctionnalités de SharePoint ainsi que de CRM afin de leur permettre la construction d'applications collaboratives ou encore de gestion de la relation client. Ces deux derniers Services, bien qu'annoncés, ne sont pas encore disponibles à ce stade.

Langages : des jumeaux et du dynamisme

La PDC fut aussi l'occasion pour Microsoft de dévoiler les prochaines versions des langages phares de la plate-forme .Net : C# et VB.Net, sans pour autant oublier les langages dynamiques qui font une entrée en force, ni



la programmation parallèle, et encore moins la prochaine étape avec F#.

C# et VB : main dans la main

Nous avons été surpris par la liste des nouveautés, des évolutions, même s'ils restent dans la droite ligne des versions actuelles, on constate une vision à long terme vers laquelle Microsoft souhaite nous amener.

Si jusqu'à présent, il existe une différence fonctionnelle entre C# et VB, la PDC, puis la TechEd développeur de Barcelone, ont confirmé que : "non VB n'est pas mort, non, C# n'est pas l'unique langage managé de .net, oui, il va y avoir une synchronisation entre deux langages". Cette synchronisation apparaît dans le fait que les équipes de ces langages vont implé-

menter les nouveautés de l'un, dans l'autre. Bref, Microsoft veut faire taire les rumeurs d'enterrement de VB ou encore sa prédilection pour tel langage ! Cela signifie que dès .Net 4.0, C# 4 et VB 10 évolueront de la même manière.

Nous ne ferons pas ici la liste de toutes les nouveautés, il faudrait y consacrer un dossier entier.

Programmation dynamique

L'autre grosse nouveauté, que les développeurs Silverlight 2 connaissent : la programmation dynamique avec les langages dynamiques de type Python, Ruby. Cela devient possible grâce à la DLR. Pour faire simple, La DLR est une CLR, mais dédiée aux langages dynamiques. D'ores et déjà les projets IronPython et IronRuby permettent cela. Mais rien n'interdit, à l'avenir, d'avoir d'autres langages.

L'ouverture de la plate-forme Mesh et Live Services aux développeurs !

L'annonce de Windows Azure comme couche fondamentale dans la stratégie " sur le nuage " a permis aux déve-



loppeurs de découvrir l'un de leurs futurs environnements de développement : " Windows Azure Services " ! Cette nouvelle plate-forme de développement apporte un concept proche des habitudes du développeur .NET : la notion de classes unifiées qui se voit transférée vers une notion de services unifiés. L'éditeur a dévoilé 5 services (disponibles depuis la PDC en CTP, pré-version que vous retrouverez tout ou partie sur notre CD-Rom du mois ou sur notre site !).

Cela signifie que dès aujourd'hui vous pouvez tester, développer la plate-forme et l'ensemble des briques de services offerts ! C'est en effet important car Microsoft a bien précisé que de nombreuses fonctions restent à implémenter et le feed back des développeurs est important pour faire évoluer les services d'ici la version finale. D'autre part, durant la TechEd 2008 à Barcelone, nous avons pu connaître une autre " zone " du modèle de programmation : le building blocs " Live Services ".

Un démarrage immédiat

Pour pouvoir démarrer la programmation Azure, il suffit de récupérer une clé d'activation, et après référencement sur la plate-forme d'enregistrement (<http://lx.azure.microsoft.com/>), on devient développeur Live !

Les Live Services vous permettront d'accéder de manière programmatique tant aux services de la plate-

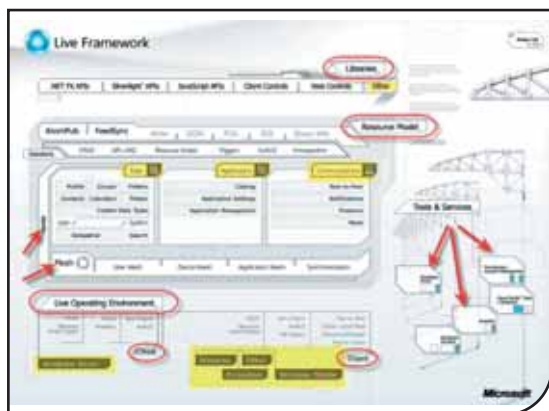
forme Live (Graphe social, Gestion des contacts, Indicateur de présence, Recherche Live, Géo localisation, ...) qu'à vos informations et données exploitées au travers de Live Mesh. Non seulement vous disposerez d'un Live Services SDK exploitable sous forme d'assemblies à ajouter à tout type de solution .NET, mais vous aurez également la possibilité de développer des applications disponibles directement dans votre interface Mesh Live Desktop ou depuis votre bureau en mode externalisé de votre navigateur.

Ainsi, Microsoft poursuit son travail d'uniformisation du développement de sa plate-forme en apportant aux développeurs .NET (de préférence ASP.NET, AJAX ou encore Silverlight) la possibilité de créer facilement et rapidement des applications dites " On the :Cloud_ " mais également " On the :Client_ "

La démarche de création d'application est ainsi des plus simplifiées et le déploiement d'applications Mesh se fera au travers d'un package extrait depuis Visual Studio 2008 SP1.

L'ajout d'une application dite Mesh à votre Live Desktop se fera de façon proche du mode exploité par la plate-forme FaceBook. Ajoutez, supprimez, partagez vos applications Mesh et leur exploitation comme vous le souhaitez ! Quelques liens :

- Azure Services Platform : <http://msdn.microsoft.com/en->



us.azure/default.aspx

- Plate-forme de développement Live :
<http://dev.live.com/>

- MSDN Azure :
<http://msdn.microsoft.com/en-us/library/dd179367.aspx>

- Plate-forme Mesh pour développeurs :
<https://developer.mesh-cpt.com/Welcom/default.aspx>

Windows 7 : faire oublier Vista !

Il a fallu attendre le deuxième jour de la PDC pour rentrer dans le vif du sujet, avec du très très lourd : Windows 7, Oslo, .net 4, Visual Studio 2007, Silverlight sur le desktop ! Ray Ozzie fit un rapide historique de la plate-forme depuis son origine jusqu'à Azure...

Il expliqua que nous vivons une période et époque formidable où l'internet va quitter peu à peu l'ordinateur tel que nous le connaissions afin de prendre son envol et de dépasser ses limites actuelles... le navigateur !

A terme, selon le VRP vedette, nous vivrons Internet comme une forme de méga accès ubiquitaire à nos données, de par cette synergie entre tous les types de terminaux qui nous entoureront... c'est d'ailleurs l'un des challenges internes relevés par Microsoft au travers de l'évolution de sa plate-forme vers Azure, de son Windows 7 et enfin de Windows Live de manière conjointe à Windows 7.

Windows 7 n'est certes pas la révolution attendue par beaucoup. Mais il constitue une évolution sérieuse de notre Windows actuel. Citons parmi les nouveautés phares :

- Gestion de l'interface de votre bureau
- Gestion de la barre des tâches et évolution de son mode preview
- vous avez par exemple la possibilité maintenant de bouger les tâches en cours dans la barre...
- Amélioration de la recherche
- Evolution des librairies
- Amélioration de la recherche avec mise en *highlight* des éléments recherchés
- Nouveau player pour le média player

- avec amélioration de gestion des médias au travers de l'ensemble des devices de votre réseau

- Optimisation de devices connectés à votre PC
- Placez les gadgets où vous souhaitez sur votre bureau
- Gestion de la personnalisation de votre bureau avec des effets personnalisés améliorés (glass color)
- Gestion des icônes de notification
- optimisation de la sécurité
- Gestion du multi-touch avec des modes propres à la gestion par la main/doigts
- Nouveaux Wordpad, calculator, Paint, ...
- Intégration de fonctionnalités VHD.

Bien que toutes ces nouveautés puissent être des plus attrayantes et plus particulièrement la notion de montage de VHD virtuels pour nous développeurs... imaginez vous promener avec votre système sur une clé USB et vous connecter depuis n'importe quelle machine (y compris en mode boot), Toutefois, après l'effet magique, la raison nous pousse à conclure que nous ne retrouvons pas de réelles avancées majeures au regard de Windows Vista...

Windows 7 aurait-il un petit goût de Vista corrigé ? Une phrase résume la pensée de Microsoft : " transition de Windows Vista "...

Un message a été martelé : la cohabitation, presque intimiste, de Win7 et de Windows Live.

Concrètement, Win7 sert de cœur à la plate-forme, associé aux Windows Live Essentials (applications riches) et aux services Live. Bref, on fait du SaaS / Software + Services même sur son PC !

Support pour le développeur au sein de Windows 7

Win7 n'a pas oublié le développeur. Et là aussi les nouveautés seront nombreuses pour pouvoir intégrer dans ses applications les fonctions proches de Win7.

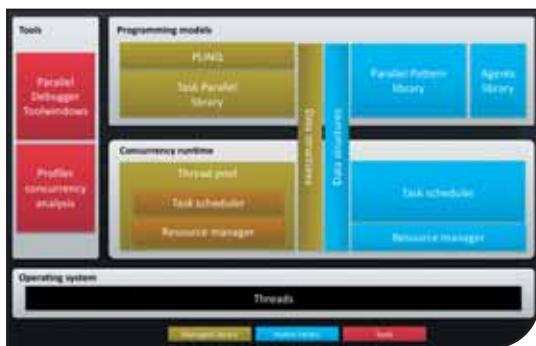
Les principales nouveautés, mises en avant, sont le ruban à la Office 2007, les " jump lists ", les nouvelles librairies,

le multi-touch, l'encre numérique, la voix et la mise à niveau de DirectX ! Si Visual Studio 2008 et .Net 3.5 fonctionneront avec Win7, le modèle de développement est cependant : Visual Studio 2010 + .Net 4.0 + les SDK Win7.

Les fondamentaux de Seven

Pour cette nouvelle évolution de Seven, Microsoft met en avant : Sécurité, rapidité, gestion de l'énergie On bénéficiera d'une bonne gestion des disques durs virtuels (ou VHD). Ainsi une simple clé USB peut devenir un VHD sur lequel on pourra booter ! La gestion des écrans sera affinée ainsi que de nouveaux mécanismes de zoom, etc. La volonté semble réellement tournée vers la maîtrise de l'utilisateur, voire du développeur, de sa machine Windows 7 et pour





cause, vous pourrez par exemple y manager vos messages d'erreur (affichage ou non).

Ainsi, Microsoft a martelé durant la PDC les éléments suivants pour le développeur :

- dès aujourd'hui installer la pré-version
- développer directement en 64-bits
- se focaliser sur les fondamentaux de son code
- intégrer ses applications le plus finement possible avec le desktop Win7
- évaluer, tester les nouvelles API
- utiliser dès maintenant les standards inclus dans Internet Explorer 8
- Charger et installer la bêta de Windows Live !

Visual Studio 2010 par Scott Guthrie

Durant la PDC, nous avons pu remarquer la forte présence de Scott Guthrie, un des plus influents membres Microsoft de l'ère Post-Gates. Avec l'apparition de Windows 7, le développeur bénéficiera de grosses nouveautés. On pourra utiliser des kits de contrôles facilitant le plus possible le développement Win7, avec notamment des contrôles de type datagrid, datapicker, calendrier,

ruban, visual state manager, etc. Autre très grosse annonce : Visual Studio 2010 est basé sur WPF ! C'est la première fois que l'IDE de Microsoft utilise la librairie d'interface apparue avec .Net 3.0. Et comme Visual Studio est un outil hautement sensible (stabilité, performance), cela permettra d'éprouver WPF. Le modèle de développement Win7 demeure profondément : .Net et Win32 (même si nous sommes de plus en plus dans un modèle 64-bits). Et pour les développeurs C++, les MFC subsistent une mise à niveau majeure.

Surtout, l'interopérabilité entre code natif et managé sera largement revue et améliorée. Les langages dynamiques font aussi leur entrée en force dans .Net (grâce au nouveau DLR de Silverlight). Cela signifie que l'on peut coder des applications .Net en Ruby si on le souhaite ! Autre bonne nouvelle : le modèle composant a été étendu. Et ce n'est pas fini ! Avec Visual Studio 2008, Silverlight 2, .Net 4, Expression Studio, il s'agit de couvrir l'ensemble des types d'application et de terminaux : web, desktop, serveurs, en ligne (cloud), mobile ! Le tout avec un modèle de développement le plus uniforme possible !

Le développement web subit une sérieuse amélioration avec la présence d'un designer Silverlight, de jQuery, d'ASP.Net 4.0, d'outils pour Ajax et Javascript et surtout d'importantes avancées sur le multicore et le développement parallèle.

Bilan : où va Microsoft ? Quelle vision ?

Nous pourrions prendre le rôle de Madame Irma et tenter de répondre par la prospective. Le seul point factuel que nous retiendrons à ce stade avec l'annonce de Windows Azure est que Microsoft se repositionne dans la course du " on the cloud " lancée par Google avec les Apps Engine , voire même prend une large avance de par l'historique de la plate-forme .NET, l'aboutissement actuel ainsi que la stratégie tant technique que fonctionnelle de ce nouveau genre d'OS !

Ce qui est indéniable, c'est que Micro-

soft se donne également les moyens de déployer sa stratégie du " on the Cloud " avec déjà un Data Center sur la côte Est des Etats-Unis et un second sur la côte Ouest, annoncé à la PDC, et un futur proche avec des Data Center Azure éparpillés sur la planète. Ce déploiement de moyens s'est également fait ressentir au travers de produits bien aboutis pour une PDC dont, en définitive, nous n'attendions qu'une vision et des annonces de stratégies de développement.

Microsoft pourrait ainsi booster, voire relancer un marché montant hésitant, en convainquant et rassurant de nombreux acteurs petits ou grands à partir dans les nuages ! La stratégie Microsoft sur les 10 prochaines années semble toute dessinée, car comment résoudre la problématique de licensing à l'échelle mondiale sinon au travers du " On the Cloud " ?

Effectivement, non contents de suivre les sessions plénières, de la PDC, afin de vous relater l'entièreté des nouveautés annoncées, nous avons pris le temps de tester par nous-mêmes dans les labos, les SDK, les technologies présentées. Et ce premier contact est plus que positif.

Les impacts se feront plus ressentir sur des notions d'architecture et de maîtrise d'ouvrage et d'outils ainsi que dans les nouveaux services mis à disposition par Windows Azure et consorts ! Préparez-vous à revoir la structuration (types de profils) de vos équipes de projets devant incorporer une stratégie Azure : prévoyez aisément des profils de type SharePoint ou encore d'expertise XML / Technologies Web... ne négligez pas non plus un bon profil sénior maîtrisant la plate-forme .NET existante et ayant une bonne lecture des impacts de mise en œuvre de solutions dites " sur le cloud " !



■ Gregory Renard
MVP, Region
Director CTO et
en charge de la
Recherche & Innovation de Wygwan

Visual Studio 2010 et .NET 4.0 : tour du propriétaire !

Sur le CD
du numéro !



La prochaine version annoncée de Visual Studio aura pour ambition de démocratiser la gestion du cycle de vie des applications (ALM – Application LifeCycle Management), améliorer la fluidité des processus de développement, rapprocher la vision “ fonctionnelle ” de celle de la “ technique ”, sans oublier la programmation parallèle, le support de Windows 7, une mise à jour commune des langages .Net, toujours plus de dynamique... Bref, le meilleur de Visual Studio en 2009 !

Actuellement disponible en CTP (pré-version technique), elle s'exécute uniquement dans une machine virtuelle (Virtual PC). Rien que l'archive à télécharger pèse plus de 7 Go ! Et ne nécessite pas moins de 2 Go de mémoire vive, bref une machine puissante... Rappelons qu'il ne s'agit là que d'une pré-version ! Nous allons faire le tour du propriétaire.

Les outils de modélisation

Grâce à l'architecture de Visual Studio Team System 2010, les différents interlocuteurs d'un projet (techniques ou fonctionnels) auront la possibilité de collaborer sur la même modélisation au travers d'outils graphiques. Basés sur UML (Unified Modeling Language) ou les DSL (Domain Specific Language), ils vous permettront ainsi d'exprimer vos besoins sous la forme de diagrammes proches des représentations métier du domaine. L'explorateur d'architecture, par exemple, vous permettra de mieux comprendre la structure des composants existants au travers de graphes, diagrammes et matrices de dépendances. [Fig.1]

Les outils de modélisation de Visual Studio 2010 sont la première étape d'une vision d'ensemble nommée " Oslo ", même si Oslo est une véritable plate-forme, elle demeure pour le moment, indépendante des outils. Cette vision repose sur le fait que, dans les années à venir, le développement évoluera davantage vers de la modélisation en rapprochant les notions métiers des conceptions techniques. Une application serait alors davantage conçue selon

les expressions du domaine que d'après l'assemblage de composants techniques, ce qui permettra de réduire l'effort et les erreurs de transformation des besoins fonctionnels en équivalence de code technique. Pour le développeur, " Oslo " signifie une orientation vers le développement (et pas seulement le développement) par les modèles, et donc des outils de modélisation qui se feront de plus en plus présents dans Visual Studio, mais aussi d'autres produits de la plate-forme .NET. Et le support annoncé des diagrammes UML s'annonce comme une étape importante dans l'univers Windows / .net.

Amélioration des outils de test.

Vs 2010 améliore là aussi beaucoup les choses ! En effet, quoi de plus frustrant pour un développeur de ne pas pouvoir reproduire un bug trouvé par un testeur ? Grâce aux nouveaux outils intégrés à Visual Studio 2010, les bugs dits " non-reproductibles " ne le seront désormais plus. [Fig.2]

Le " **Microsoft Test Runner** " est un outil autonome qui permet de guider les testeurs au travers de différentes étapes pour compléter un cas de test. Lorsqu'un testeur démarre un nouveau cas de test, cet outil prend une " photo " des données du système (version de l'OS, service packs installés, ...) et capture ensuite des copies d'écran ou vidéos permettant de mieux comprendre les facteurs qui ont amené à la découverte d'un bug. Lors de la création d'un bug dans Team Foundation Server par le testeur, l'ensemble de ces informations sont attachées, permettant ainsi au dévelop-



Fig.1

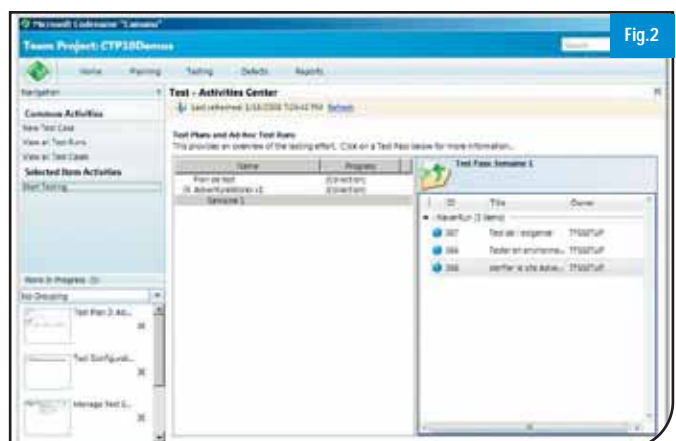


Fig.2

peur en charge de la correction d'être plus efficace et d'avoir un maximum d'informations sur les conditions de reproduction du bug. La " **Test Impact View** " permet de ne tester que la partie de code modifiée. Cette vue permet au développeur de voir la liste des tests qui sont impactés par la modification de code. [Fig.3]

Amélioration de Team Foundation Server.

Dans cette nouvelle version, les équipes de développement pourront plus facilement adapter les méthodologies qu'elles souhaitent utiliser pour leurs projets. Le suivi de la réalisation se trouve amélioré par la mise en place d'une hiérarchie entre " work items " liés. De nouveaux outils visuels permettront de suivre les évolutions de code entre différentes branches (développement & production par exemple). De nouvelles possibilités de gestion des builds grâce à l'introduction de workflows permettant de gérer les erreurs avant qu'elles n'impactent le reste de l'équipe. Et de manière générale, des améliorations notables pour l'administration et le déploiement de la partie serveur de Visual Studio Team System.

Découverte de VS 2008 [Fig.4]

Pour commencer à utiliser la version 2010 de Visual Studio, Microsoft met à disposition une Community Technology Preview sous la forme d'un PC virtuel (Virtual PC 2007 SP1). Prévoir au moins 2Go de RAM et 30Go de disque dur pour le VHD. L'image virtuelle vient avec une application exemple (DinnerNow) et un ensemble de documents permettant de dérouler des exemples mettant en avant les nouvelles fonctionnalités de Visual Studio 2010. WPF (Windows Presentation Foundation) fait son apparition dans Visual Studio et l'extensibilité devrait reposer dorénavant sur MEF (Managed Extensibility Framework). Ce framework fournit des mécanismes de découverte et de composition qui permettent d'étendre très simplement l'IDE de Visual Studio. Une " promesse " qui devrait ouvrir la voie à de nouveaux scénarii très intéressants. [Fig.5]

.NET 4.0 : un framework parallèle ?



Fig.3

Bien que peu de détails aient été publiés sur le .NET Framework 4.0, les informations publiques montrent que le .NET Framework 4.0 met l'emphasis sur le support des technologies parallèles qui ciblent les ordinateurs multi-cœurs et les systèmes distribués.

Suite à la PDC, le **poster** du framework 4.0 a été publié et une version DeepZoom est disponible : http://brad_abrams.members.winisp.net/Projects/PDC2008/DotNet4Poster/DotNetFramework4PosterDeepZoom.htm

Une présentation de la PDC dédiée à la programmation multi-cœur a donné une vision schématisée des outils et nouveaux modèles de programmation. Et VS 2010 marque l'arrivée en force du développement multi-core parallèle dans la programmation .Net et au quotidien ! [Fig.6]

PLINQ (Parallel LINQ)

PLINQ permet d'exécuter des requêtes concurrentes LINQ afin de paralléliser les requêtes sur les objets (LINQ to Objects) et sur les données XML (LINQ to XML). La nature déclarative de LINQ permet naturellement et facilement de paralléliser les traitements exprimés. PLINQ est très intéressant dans les traitements de données intensifs utilisant de multiples sources de données. PLINQ utilise les mêmes mots clés que LINQ (OrderBy, Join, Select, Where...) et permet de requêter les objets de type T[], List<T> et tout type dérivant de IEnumerable<T> en plus des documents Xml chargés par les APIs System.Xml.Linq. Par exemple, le code suivant permet de filtrer une collection de données :

```
IEnumerable data = new int[] {1, 2, 3, 4, 5, 6};
var q = from i in data.AsParallel()
        where i > 4
        orderby i
        select i;
foreach(var i in q) ...
```

L'exécution des différentes clauses est automatiquement parallélisée par PLINQ, via plusieurs threads et le tout n'est exécuté qu'au moment de l'accès aux données via foreach. Bien que cela semble simple de paralléliser les requêtes LINQ sur les objets, il ne faut pas oublier les principes liés aux traitements par plusieurs threads. Exemple :

```
int i = 0;
var q = from n in numbers select ++i;
```

L'incrément de la variable " i " est réalisé en parallèle et de manière non atomique, ce qui peut engendrer des résultats inattendus...

Task Parallel Library (TPL)

Cette librairie permet de paralléliser l'exécution de boucles (for ou while) avec l'utilisation de méthodes classiques et de delegates. La librairie prend en charge la répartition des traitements sur plusieurs threads. Par exemple le code suivant :

```
for (int i = 0; i < 100; i++) {
```

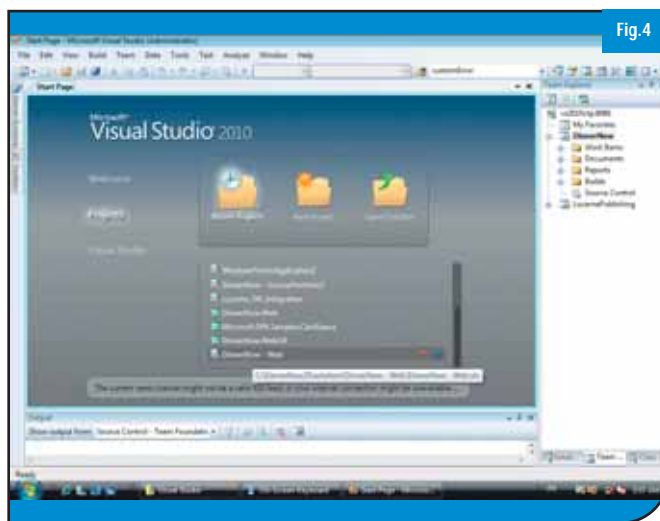


Fig.4

```
a[i] = a[i]*a[i];
}
```

se parallélise de la manière suivante (les calculs de chaque élément du tableau sont indépendants des autres) :

```
Parallel.For(0, 100, delegate(int i) {
    a[i] = a[i]*a[i];
});
```

De plus, la librairie inclut aussi des types tels que " Task " ou " Future " : " Task " est une action qui peut-être exécutée de manière asynchrone, tout comme un thread, sans que le développeur ait à se soucier de synchronisation. " Future " est une tâche spécialisée qui retourne un résultat. Le résultat est calculé par un thread encapsulé par un objet " Future " et est stocké dans un buffer jusqu'à sa récupération.

Sans oublier la ... DLR

La Dynamic Language Runtime, ou DLR, permet aux langages dynamiques, tels que IronPython et IronRuby, de faire leur apparition sur la plate-forme .NET et aux langages existants d'intégrer du dynamisme. Jusqu'à présent la DLR était présente dans Silverlight 2.0. Les résolutions dynamiques de types sont prises en charge par la DLR qui offre, entre autres, des fonctionnalités de mise en cache pour l'amélioration des performances. La plupart du temps, la DLR restera invisible au développeur mais pour interagir avec elle, il suffira d'implémenter l'interface IDynamicObject afin d'intégrer son propre comportement. L'arrivée de la DLR est une réelle avancée pour les développeurs, comme ce fut le cas dans la plate-forme Java il y a quelques mois ! [Fig.7]

Et d'autres langages devraient compléter rapidement l'offre ! Rappelons aussi que le projet DLR est sur codeplex et que .Net 4 intégrera la v1. Comme tout bon langage .Net, il est possible de mixer langage dynamique et statique. A la PDC, un exemple éloquent fut donné :

```
static int Noddy_init(Noddy *self, PyObject *args, PyObject *kwargs)
{
    PyObject *first=NULL, *last=NULL, *tmp;

    static char *kwlist[] = {"first", "last", NULL};
```



Fig.5

```
if (! PyArg_ParseTupleAndKeywords(args, kwds, "|OO", kwlist,
                                &first, &last))

    return -1;

if (first) {
    tmp = self->first;
    Py_INCREF(first);
    self->first = first;
    Py_XDECREF(tmp);
}

if (last) {
    tmp = self->last;
    Py_INCREF(last);
    self->last = last;
    Py_XDECREF(tmp);
}

return 0;
}
```

Autre exemple :

```
dynamic calc = GetCalculator();
int sum = calc.Add(10, 20);
```

Là, on insère directement un type dynamique en C# 4... Bref, que du bonheur pour le développeur. Et surtout vous pouvez construire votre propre librairie dynamique.

Retenons aujourd'hui les éléments suivants :

- intégration à .Net
- combinaison des langages .Net
- utilisation des outils .Net, avec la même expérience

Les nouveautés côté langages...

C# 4.0 : Quel dynamisme !

L'introduction du dynamisme dans le langage C# devrait être l'évolution majeure. Par dynamisme, nous entendons des structures de données inconnues lors de la compilation et générées dynamiquement à l'exécution.

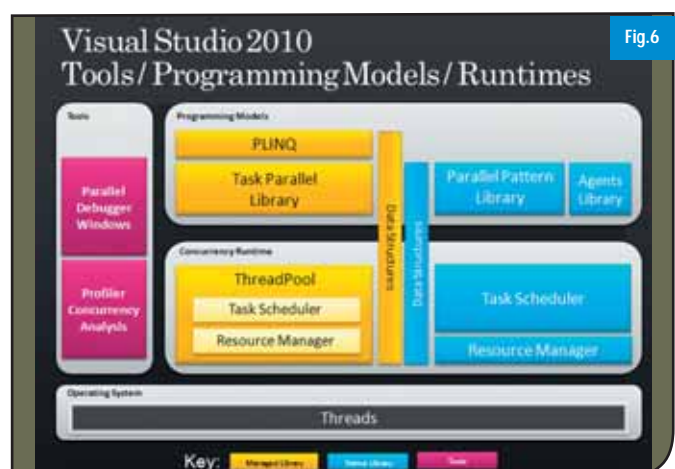


Fig.6

- **Résolution dynamique de type ou Dynamic Lookup** : une structure dynamique (mot-clé `dynamic`) permet de vérifier le type à l'exécution et non plus à la compilation. Le développeur pourra donc écrire du code qui, en temps normal, ne compilerait pas car la résolution de méthodes, indexeurs, opérateurs, propriétés, etc. se fait à l'exécution.
- **Paramètres nommés et optionnels** : Dans cette nouvelle version du C#, le développeur pourra donner des valeurs par défaut aux paramètres d'une méthode et par conséquent les rendre optionnels lors de l'appel. De plus, les paramètres pourront être passés par leur nom et plus seulement par leur position.
- **Interop COM** : de par l'apport des 2 points précédents, l'interopérabilité COM se voit simplifiée : résolution dynamique de type et paramètres optionnels et nommés, plus quelques améliorations promises.
- **Variance dans les types génériques** : Pour résumer cette nouveauté, rien de mieux que le code ci-dessous pour comprendre ce que cela veut dire :

```
List<int> ints = new List<int>();
ints.Add(1);
ints.Add(10);
ints.Add(42);
List<object> objects = new List<object>();

objects.AddRange(ints); -> Ce code ne compile qu'en C# 4.0!
```

VB : 10e du nom et unification vers le haut du langage !

Quel chemin parcouru depuis 1991 pour ce langage et ce numéro de version tout rond pourrait sonner comme une date d'anniversaire. Pour cette version, le mot clé est : productivité !

- **Lambdas multi-lignes ou en ligne** : Et voici quelques exemples explicites qui se passeront d'explications :

```
Dim nums() As Integer = {1, 2, 3, 4, 5}
nums = Array.FindAll(nums, Function(n)
    Console.WriteLine("testing " & n)
    Return n > 2
End Function)
```

Ou encore

`Array.ForEach(nums, Sub(n) Console.WriteLine(n))`

- **Continuité implicite d'une ligne de code** : Le caractère " _ " ou underscore est utilisé lorsque qu'une ligne de code est séparée sur plusieurs lignes. Avec VB 10.0, la plupart du temps, le déve-

loppeur n'aura plus à se soucier d'utiliser l'underscore pour signifier que son code continue à la ligne suivante.

- **Propriétés auto-implémentées** : Plus besoin d'implémenter le détail d'une propriété, sa déclaration suffira au compilateur pour créer automatiquement son implémentation. Ainsi le code suivant :

```
Private _FirstName As String

Property FirstName() As String
Get
    Return _FirstName
End Get
Set(ByVal value As String)
    _FirstName = value
End Set
End Property
```

Se simplifie par l'écriture suivante :

```
Property FirstName() As String
```

Et il est possible de spécifier l'initialisation qui sera automatiquement implémentée dans le constructeur du type tel que :

```
Property ID() As Integer = -1
Property SupplierList() As New List(Of Supplier)
Property OrderList() As New List(Of Order) With {.Capacity = 100}

<DefaultValue("-")>
Property Name() As String Implements ICustomer.Name
```

- **Les initialisations de collections** : Le développeur pourra initialiser une collection sur la même ligne que sa déclaration. Exemple :

```
Dim list = New List(Of String) From {"abc", "def", "ghi"}
```

- **Tableaux de littéraux** : L'inférence de type est laissée au compilateur. Exemples :

```
Dim a = {1, 2, 3} 'Tableau d'entier/Integer()
Dim d = {1, "123"} 'Tableau d'objet/Object()
```

- **Paramètres optionnels sans valeur (Nullable)** : Les paramètres optionnels peuvent être initialisés ou mis à `Nothing`. Exemple :

```
Sub Add(x As Integer, y As Integer, Optional z As Integer? = Nothing)
```

Et encore d'autres nouveautés et exemples de code à découvrir sur <http://code.msdn.microsoft.com/vbfuture> pour VB et <http://code.msdn.microsoft.com/csharpfuture> pour C#.

Il y aurait encore de très nombreuses nouveautés à vous montrer. Au cours des prochains numéros, nous aurons le temps et l'occasion d'approfondir tous ces éléments !



■ Frédéric QUEUDRET

CTO de la société Mpoware, accélérateur d'innovation. Société française d'édition de logiciels et de prestations de services sur la création d'outils de productivité pour les développeurs Visual Studio.
<http://www.mpoware.com/>

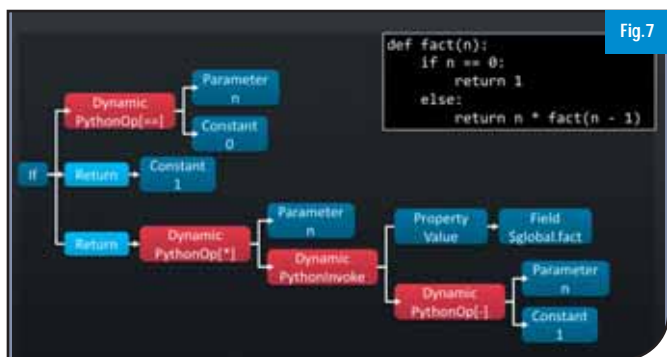


Fig.7



TechEd 2008 : Oslo, Dublin

Les TechEd 2008, se déroulant à Barcelone pour la dernière année représentèrent le temps fort des deux premières semaines de novembre. La première fut consacrée à la partie IT Pro et la seconde aux développeurs.

Juste après la PDC, riche en événements et annonces, nous attendions beaucoup de ces conférences. Las, une certaine déception se lisait sur les visages des participants, dont le nôtre. Il faut dire que les keynotes inaugurales furent particulièrement décevantes et d'un niveau technique incompréhensible, surtout suite à la PDC. Nous vous proposons cependant de plonger dans 2 outils particulièrement importants.

Oslo : la plate-forme de modélisation à tout faire !

Premier constat : Oslo est partout et fait tout ce que l'on veut, ou presque. Trois fondamentaux à retenir autour d'Oslo : workflow de Workflow Foundation, les services et les applications (pouvant combiner des workflows et des services). Et ces modèles Oslo peuvent être liés entre eux, avec des processus business, des applications, dans des workflows et certains peuvent être directement exécutables sans génération intermédiaire ni transformation ! Pour faire simple, la plate-forme repose sur un référentiel stockant les schémas des modèles, les instances. Il se compose d'un langage (langage M) et d'un outil de modélisation Quadrant, utilisable dans Visual studio ! Aujourd'hui, 6 schémas sont accessibles :

- process
- application : application composite ou non
- workflow : basé sur Workflow Foundation
- activity : une activité spécifique dans un workflow WF
- environnement : un ensemble de machines, d'applications sur lequel se déploie une application (par

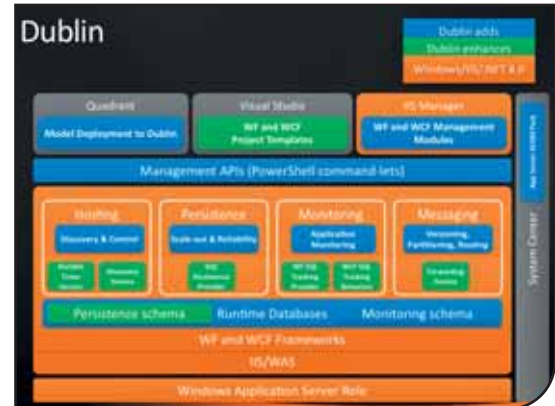
exemple sur un Datacenter virtualisé, un nuage de type Azure).

Mais Oslo n'est pas fermé, vous pouvez définir votre propre schéma de modèles. Ce qui offre une extensibilité infinie à la plate-forme et surtout une ouverture considérable vers l'extérieur ! Sans rentrer aujourd'hui dans le détail du langage M, il faut en retenir deux éléments : le MSchemar (langage définissant le référentiel utilisé par les schémas, etc.) et le MGrammar (langage textuel pour définir les DSL). Et surtout, Oslo peut s'intégrer à des outils Microsoft. Pourquoi pas dans Eclipse, ou tout autre outil de modélisation ?

Dublin

Dublin sera disponible, dans un premier temps, indépendamment mais une intégration à Windows Server n'est pas à exclure. Comment définir Dublin ? Avouons que nous avons un peu de mal mais David Chappell, un expert indépendant, a résumé à merveille la chose. Dublin permet une montée en charge sur les services WCF, intègre une persistance pour les états des services, tout en procurant des outils de gestion, un auto démarrage (pas besoin d'attendre que le premier message arrive), redémarrage des instances de services tombés. C'est très important surtout quand vous exécutez sur une longue durée des services.

On peut monitorer les services hostés via Powershell ou IIS Manager. On peut même réaliser du load balancing sur les services pour arriver à une continuité de ceux-ci. Dublin peut être vu comme une extension à IIS et Win-



dows Server Application Server. On peut scripter l'outil, rajouter des mécanismes de routing, de découverte (des services).

Un tout cohérent

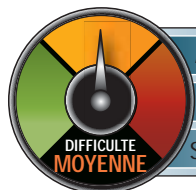
Vous l'aurez compris, Oslo est une pièce maitresse dans la plate-forme .Net / Windows / Azure / virtualisation de Microsoft. Il sert à tout, partout, pour tout modéliser. Il présente une ouverture maximale pour s'intégrer avec l'extérieur et son interaction avec un outil comme Dublin ouvre de nouveaux scénarios de déploiement / administration sans égal sous Windows ! Bref, une plate-forme à regarder de très près. Nous sommes convaincus que la modélisation (UML, non UML, DSL) est le nouveau leitmotiv Microsoft pour les développeurs, les entreprises, les administrateurs, les DSI !

Nous verrons très prochainement les autres annonces et nouveautés de la PDC / TechEd comme le CCR, Gemini, C++, etc. Nous avons presque un an pour vous expliquer, décortiquer, coder avec tous ces outils et technologies qui changent en profondeur l'univers du développeur .Net et Windows

■ François Tonic

facebook : soyez OpenSocial !

Facebook avait ouvert le bal en proposant aux développeurs de créer leurs propres applications. Google a suivi le pas en proposant une API qui pourrait être implémentée par tous les réseaux sociaux qui le voudraient : OpenSocial. API qui ne cesse de s'étendre !



APPLICATION : SOCIAL

LANGAGE : XML

SOURCE : NON

Depuis le lancement d'OpenSocial, de nombreux nouveaux réseaux, appelés "conteneur", sont venus s'ajouter à la liste impressionnante des sites de network

ing compatibles dans lesquels on retrouve par exemple MySpace et Orkut. Pour consulter la liste complète, rendez-vous à cette url : <http://code.google.com/apis/opensocial/gettingstarted.html>

LE PRINCIPE DE FONCTIONNEMENT

Une application OpenSocial ou Gadget est une sorte d'extension pour le réseau social dans lequel vous allez l'ajouter. Cette application est hébergée sur un serveur (le vôtre par exemple) mais vous pouvez accéder à quelques informations en provenance du réseau "conteneur". Typiquement, vous pouvez récupérer l'identifiant de l'utilisateur connecté à votre application. Lorsqu'un utilisateur va ajouter votre application, le conteneur devra parcourir le code de celle-ci afin de l'afficher. En effet, la définition de l'interface se fait à l'aide d'un meta-langage basé sur XML. De la même manière, vous avez Facebook qui propose de décrire l'interface des applications à l'aide de FBML. Voici un exemple basique d'application OpenSocial :

```
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
  <ModulePrefs title="Hello World!">
    <Require feature="opensocial-0.8" />
  </ModulePrefs>
  <Content type="html">
    <![CDATA[
      Bonjour Programmez!
    ]]>
  </Content>
</Module>
```

QU'EST CE QU'UN CONTENEUR ET LEQUEL CHOISIR?

Le réseau social qui héberge l'application s'appelle le conteneur. Bien entendu, OpenSocial ne fonctionne pas sur tous les réseaux. Il faut que le réseau en question l'implémente. Heureusement pour nous, il y a pas mal de conteneurs. Vous retrouverez une liste non exhaustive de ceux-ci dans le tableau ci-dessous :

Container	OpenSocial API	Getting Started Docs	Status
hi5	v0.8	hi5 Developer Site	Tous les utilisateurs
Webon	v0.8	Webon team blog	Tous les utilisateurs
Freebar	v0.7	Freebar Developer Zone	Tous les utilisateurs
Friendster	v0.7	Friendster Developers Platform	Tous les utilisateurs
IDtail	v0.7	IDtail developer site	Tous les utilisateurs
MySpace	v0.7	MySpace Developer Site	Tous les utilisateurs
Ning	v0.7	Ning Developer Network	Tous les utilisateurs
orkut.com	v0.7	orkut Developer Home	Tous les utilisateurs
YiQi	v0.7	YiQi Developer Home	Tous les utilisateurs
lokalisten	v0.8	lokalisten developer platform	Ouvert avec sandbox
Hyves	v0.7	Hyves OpenSocial Site	Ouvert avec sandbox
iGoogle	v0.7	iGoogle Developer Home	Ouvert avec sandbox
imeem	v0.7	imeem Media Platform Home	Ouvert avec sandbox
Mail.ru	v0.7	Mail.ru Developers Page	Ouvert avec sandbox
Netlog	v0.7	Netlog OpenSocial Site	Ouvert avec sandbox
Viadeo	v0.7	Viadeo OpenSocial site	Ouvert avec sandbox
CityIN	v0.7	CityIN main site	Demande de validation
Tianya	v0.7	Tianya main site	/
Shindig sample	v0.7	Shindig+Caja Intro	Uniquement pour développeurs
Plaxo Pulse	v0.5	Plaxo's Gadgets Page	/

Le choix du conteneur demande de la réflexion. D'abord, car votre application doit pouvoir tourner sur n'importe quel conteneur. Il faut donc penser correctement son application afin de la rendre la plus portable possible. Ensuite, il faut bien se rendre compte qu'une application de *poke à friend* sera très bien vue sur MySpace ou hi5 mais certainement pas sur Viadeo. Il faudra donc bien regarder quelle est la cible des différents conteneurs. Il existe en effet des réseaux orientés fun – loisir et d'autres orientés business. Il faut savoir quel type d'application vous voulez créer et par conséquent choisir les bons conteneurs sur lesquels déployer votre application. Pour cela, inscrivez-vous sur les différents réseaux pouvant servir de conteneur et parcourez un peu le site pour voir quel type de population fréquente le réseau. En plus d'avoir deux gros types de réseaux, on peut encore classer en sous-catégories de réseaux. Ainsi, Viadeo est considéré comme un réseau riche en dirigeants de société. LinkedIn (qui devrait implémenter OpenSocial), lui, est un peu différent, il regroupe énormément de personnes du secteur de l'IT/Télécom. Il y a également beaucoup de RH sur LinkedIn. Sans parler de certains réseaux qui représentent, plus que d'autres, certaines régions du monde. Ainsi Viadeo montre une image assez



Les outils de la Direction Informatique

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



L'INFORMATION SUR MESURE

Actu triée par secteur

Cas clients

Avis d'Experts



Actus

Événements

Newsletter

L'INFORMATION EN CONTINU

www.solutions-logiciels.com



française, tandis que d'autres réseaux représentent plutôt les pays asiatiques ou encore les Etats-Unis.

Aujourd'hui, beaucoup de réseaux voient le jour. On les appelle des réseaux niche. Ils sont très ciblés. On en retrouve sur la thématique du sport, d'autres sur les voyages ou encore des réseaux réservés uniquement aux femmes. Ils ne font pas encore partie de OpenSocial mais ils pourraient très rapidement rejoindre ce groupe. OpenSocial évolue très rapidement et promet d'être un acteur majeur du réseautage social dans l'année à venir. Il faut s'attendre à plusieurs nouveautés dans les mois à prochains.

Google, initiateur du projet, s'intéresse fortement aux sites internet d'e-commerce. On pourrait d'ailleurs voir arriver eBay comme conteneur OpenSocial. Les possibilités sont donc énormes dans ce projet. Pouvoir ajouter des outils sur eBay ou pouvoir récupérer les informations d'utilisateurs eBay peut être très intéressant, même si aujourd'hui, c'est déjà possible à l'aide de certaines API.

MA PREMIÈRE APPLICATION

Pour pouvoir créer votre application vous avez besoin principalement de deux choses :

- Un hébergeur
- Un compte sur un des réseaux sociaux supportant OpenSocial (Orkut par exemple)

Un bloc note suffit à l'édition d'un gadget mais Google propose un Gadget Editor que vous pouvez retrouver à l'adresse suivante :

<http://code.google.com/apis/gadgets/docs/legacy/gs.html#Scratchpad>

Cet éditeur va vous permettre de prévisualiser votre gadget.

Pour installer notre premier gadget sur Orkut, vous devez faire plusieurs choses :

1. Créer un fichier xml (gadget.xml par exemple).
2. Le placer dans un dossier sur votre serveur.
3. Vous rendre sur <http://sandbox.orkut.com>.
4. Cliquer sur "edit" dans Orkut.



5. Ajouter votre application en donnant l'url de celle-ci dans la boîte prévue à cet effet.

Note : Il faut demander l'accès à Google pour cette sandbox. Cela prend généralement quelques jours.

DÉCOMPOSITION D'UN GADGET

Nous allons créer notre premier gadget OpenSocial. Il a pour but de permettre aux utilisateurs d'un réseau de s'envoyer des cadeaux. On connaît très bien ce type d'application puisque Facebook en possède une très connue du même genre. Elle n'est, normalement, pas très compliquée mais utilise plusieurs concepts. Commençons tout de même par une petite base théorique. D'abord, vous devez absolument connaître la constitution d'un simple gadget. Voici un exemple simple de gadget :

```
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
  <ModulePrefs title="Hello World!">
    <Require feature="opensocial-0.7" />
  </ModulePrefs>
  <Content type="html">
    <![CDATA[
      Hello, world!
    ]]>
  </Content>
</Module>
```



Dans cet exemple hello world, vous remarquez plusieurs sections dans le XML. Chaque section a bien entendu son utilité. L'en-tête n'a rien à voir avec OpenSocial, c'est l'en-tête que nous retrouverons dans tout type de fichier XML :

```
<?xml version="1.0" encoding="UTF-8" ?>
```

Directement après l'élément racine de notre structure XML, nous avons le nœud module. Cet élément permet d'indiquer à OpenSocial que ce fichier XML contient un gadget. Juste en-dessous de cet élément, on retrouve un élément enfant ModulePrefs. Dans cet élément, plusieurs informations permettent d'identifier l'application et son auteur. Il possède un élément fils Require qui permet de dire que nous avons besoin d'une fonctionnalité dans ce gadget. Ici on demande l'accès à OpenSocial v0.7 :

```
<Require feature="opensocial-0.7" />
```

On retrouve ensuite le contenu du gadget dans le nœud Content. Celui-ci contient du HTML dans notre cas. Le HTML doit absolument

être contenu dans `<![CDATA[...]>` pour des raisons techniques. En effet, lorsque vous voulez parser du XML, vous devez faire attention à ne pas prendre les balises HTML pour des éléments de la structure XML. Le parseur le sait lorsqu'il voit `<![CDATA[...]>`.

```
<Content type="html">
  <![CDATA[
    Hello, world!
  ]]>
</Content>
```

Ce que nous retrouvons dans Content doit être l'équivalent de ce que vous mettriez dans le body de votre page HTML.



LISTER VOS AMIS AVEC OPENSOCIAL

A première vue, si vous savez développer une application web, vous n'aurez pas trop de mal à développer une application utilisant OpenSocial, à condition de bien respecter le format de l'application.

L'objet `DataRequest` est celui que vous utiliserez probablement le plus.

Cet objet vous permet de créer des requêtes pour aller chercher de l'information sur le réseau conteneur :

- Qui est connecté ?
- Qui sont les amis de cet utilisateur ?
- ...

On peut comparer cet objet à une requête http. Pour créer un objet `DataRequest` on utilise la méthode `newDataRequest()` :

```
var req = opensocial.newDataRequest();
```

On doit ensuite ajouter des paramètres à cet objet pour définir quel sera son but (chercher des amis ?) :

```
req.add(request, key);
```

Une fois la requête préparée, vous devez l'envoyer et définir une méthode de traitement :

```
req.send(fetchHandler);
```

Exemple d'application pour lister votre liste d'amis :

```
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
  <ModulePrefs title="List Friends Example">
    <Require feature="opensocial-0.8"/>
  </ModulePrefs>
  <Content type="html">
    <![CDATA[
      <script type="text/javascript">
        /**
         * Request the OWNER and OWNER's friends.
         */
        function request() {
          var idspec = opensocial.newIdSpec({ "userId" : "OWNER",
            "groupId" : "FRIENDS" });
          var req = opensocial.newDataRequest();
          req.add(req.newFetchPersonRequest(opensocial.IdSpec.
            PersonId.OWNER), "get_owner");
```

```
req.add(req.newFetchPeopleRequest(idspec), "get_friends");
req.send(response);
};

function response(dataResponse) {
  var owner = dataResponse.get('get_owner').getData();
  var friends = dataResponse.get('get_friends').getData();
  var html = 'Friends of ' + owner.getDisplayName();
  html += '<br><ul>';
  friends.each(function(person) {
    html += '<li>' + person.getDisplayName() + '</li>';
  });
  html += '</ul>';
  document.getElementById('message').innerHTML = html;
};

gadgets.util.registerOnLoadHandler(request);

</script>
<div id="message"> </div>
]]>
</Content>
</Module>
```

LES 5 RECOMMANDATIONS POUR ALLER PLUS LOIN

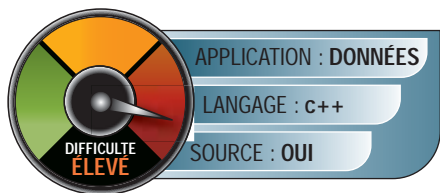
1. Bien utiliser l'objet `DataRequest` : <http://code.google.com/apis/opensocial/articles/datarequests/datarequests-0.8.html>
2. Le JavaScript : Lorsqu'il y en a peu, on prend l'option de les mettre sur la même page. Par contre, lorsqu'il commence à y en avoir beaucoup, on choisit plutôt de les mettre dans des fichiers externes. Il faut d'ailleurs faire très attention à ces fichiers car ils peuvent devenir très lourds au chargement. Certaines technologies permettent de contrôler ce poids en ne chargeant que les fichiers nécessaires au chargement de l'application et en allant rechercher le reste de façon asynchrone. Ce sont des techniques dont vous devrez tenir compte lorsque vous développerez des applications importantes. Utiliser OpenSocial avec une API de géo-localisation, est déjà lourd. Ajoutez à cela quelques API d'effets visuels et votre application devient carrément lente.
3. OpenSocial et JSON : JSON est très utilisé dans Facebook et OpenSocial car nous retrouvons beaucoup de JavaScript dans les différentes API et notamment celles de OpenSocial. En effet, JSON est un format de données qui utilise la notation des objets JavaScript. C'est en quelque sorte le format de sérialisation des objets JavaScript (si on peut parler ainsi). Vous avez sans doute remarqué que nous devons passer nos données en JSON dans le `DataRequest`. Cela nous permettra de récupérer un tableau JavaScript par la suite.
4. Voir un exemple d'utilisation de JSON par OpenSocial : <http://blogs.codes-sources.com/loicbar/archive/2008/06/10/opensocial-et-wcf-rest-ful.aspx>
5. Bien utiliser les données persistantes : <http://code.google.com/apis/opensocial/articles/persistence-0.8.html>
6. La méthode `MakeRequest` : <http://code.google.com/apis/opensocial/articles/makerequest-0.8.html>

A vous de jouer !

■ Loïc Bar

Ultimate++ et les bases de données

Encore peu connu, Ultimate++ n'en est pas moins un IDE C++ formidable. Nous découvrons aujourd'hui ses talents pour le travail avec les bases de données.



Dans le n°104, nous vous avions présenté les qualités d'Ultimate++. Aujourd'hui, nous allons découvrir ensemble comment Ultimate++ est original et puissant

pour le travail avec les bases de données. La seule chose que l'on puisse lui reprocher est une documentation pas à la hauteur quand elle n'est pas inexistante. Nous espérons que cet article remédiera un peu à cette lacune et contribuera à donner à Ultimate++ la notoriété qu'il mérite. Pour les exemples accompagnant cet article nous avons travaillé avec PostgreSQL, SGDBR favori de votre serveur. Cependant, l'adaptation à un autre système comme Oracle ou MySQL ne posera aucune difficulté, et c'est d'ailleurs un des atouts maîtres d'Ultimate++ dans ce domaine. Nous supposons que, quel que soit le SGDBR choisi, une base de données vide, du nom de UppProgrammezDB, a été créée en préalable à l'exécution des exemples. Enfin, nous partons du principe que le lecteur a lu l'article de Programmez! 104 évoqué plus haut et est un minimum familiarisé avec l'organisation et la philosophie d'Ultimate++

1 SE CONNECTER À UNE BASE DE DONNÉES

Nous commençons par travailler avec des applications console, mais utilisant les librairies d'Ultimate++. Nous avons placé tous nos exemples dans un assembly du nom de DemoPostgreSQL. Donc notre application DemoConnexion fait partie de l'assembly et a été créée comme une application de type 'Core console project', tel qu'illustré. [Fig.1]. Faisant de la paresse une vertu, nous avons demandé la génération automatique d'un en-tête et notre projet sera donc constitué de deux fichiers: DemoConnexion.h et DemoConnexion.cpp. Pour compiler nos exemples, Ultimate++ a besoin d'accéder aux en-têtes et fichiers de librairie du SGDBR utilisé. Pour cela, nous devons configurer les 'Build Methods', ou méthode de construction de projet d'Ultimate++. Celui-ci propose des méthodes de construction pour tous les compilateurs C++ qu'il a recensés lors de son installation. Pour atteindre l'assistant de configuration des

méthodes de construction sélectionnez, dans le menu d'Ultimate++, Setup|BuildMethods. Une fois dans l'assistant, veillez à bien sélectionner le compilateur que vous utilisez, puis sous les onglets 'INCLUDE directories' et 'LIB directories', cliquez avec le bouton de droite et sélectionnez 'Append row', autrement dit, ajout d'une ligne, et ajoutez respectivement [Fig.2]

C:\PostgreSQL\8.2\include

et

C:\PostgreSQL\8.2\lib

Remarque : si vous utilisez le compilateur de Visual Studio, celui-ci va naturellement chercher des librairies au format Microsoft. Sous PostgreSQL celles-ci se situent dans un sous-répertoire du nom de 'ms'. Cependant, donner C:\PostgreSQL\8.2\lib\ms ne fonctionnerait pas car Ultimate++ est intelligent et ajoute de lui-même le suffixe ms lors de l'édition de liens. Les répertoires cités plus haut sont donc valables quel que soit le compilateur utilisé. Mais bien entendu, ils devront être adaptés si vous utilisez un autre système de base de données ou bien une autre version de PostgreSQL. Cette opération n'est à faire qu'une fois et elle s'appliquera à tous les exemples. Ensuite, il faut ajouter des packages au projet. Le terme package, propre à la terminologie Ultimate++ peut prêter à confusion. Ce qu'il faut comprendre est tout simplement "librairie". En l'occurrence, nous devons ajouter les librairies SQL, librairie générique, et PostgreSQL, librairie spécialisée pour PostgreSQL. Pour cela, cliquer avec le bouton droit de la souris sur le package constitutif du projet, puis choisir 'Add package to...' dans le menu surgissant. [Fig.3] Contrairement à la configuration des 'Build methods', l'ajout des packages/librairies doit être fait pour chaque projet. Nous sommes maintenant en mesure d'écrire du code. Pour établir une connexion, c'est tout simple. D'abord le contenu du fichier en-tête :

```
#ifndef _DemoConnexion_DemoConnexion_h_
#define _DemoConnexion_DemoConnexion_h_

using namespace Upp;
#include <PostgreSQL/PostgreSQL.h>
```

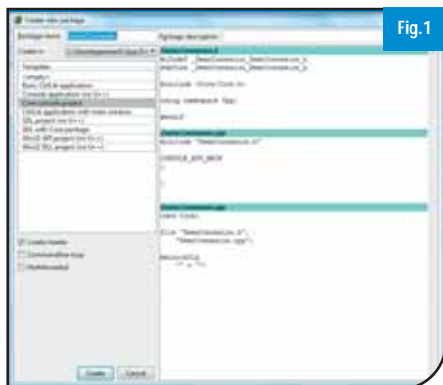


Fig.1

Création d'une application de type 'Console core' dans l'assistant d'Ultimate++.

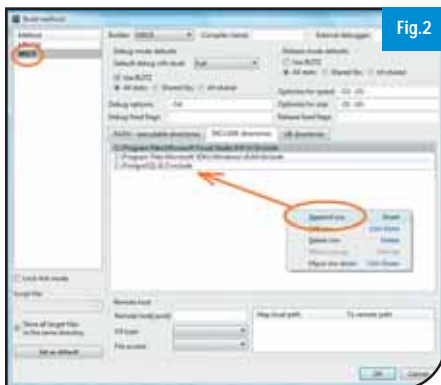


Fig.2

Configuration des build-methods pour accéder aux en-têtes et librairies du système de base de données.

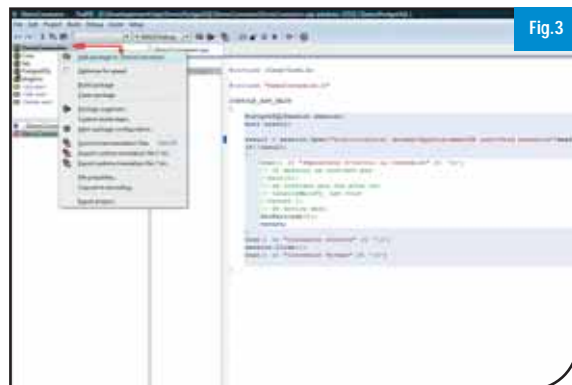


Fig.3

Ajout de packages/librairies à notre projet.

```
#endif
```

dans lequel on remarquera l'usage de l'espace de nom Upp, espace dans lequel résident les librairies d'Ultimate++. Puis le code proprement dit :

```
#include <Core/Core.h>
#include "DemoConnexion.h"

CONSOLE_APP_MAIN
{
    PostgreSQLSession session;
    bool result;

    result = session.Open("host=localhost dbname=UppProgrammezDB
user=fred password=xxxx");
    if(!result)
    {
        Cout() << "Impossible d'ouvrir la connexion" << '\n';
        // Ci dessous ne convient pas
        //exit(1);
        //return 1; ne convient pas non plus
        // car ConsoleMainFn_ est void
        // On écrira donc
        SetExitCode(1);
        return;
    }
    Cout() << "Connexion ouverte" << '\n';
    session.Close();
    Cout() << "Connexion fermee" << '\n';
}
```

Ce code appelle une seule remarque. Toute application bien élevée se doit de retourner une valeur bien définie au système d'exploitation. En cas de bon déroulement de l'application, le framework se charge de retourner 0. En cas d'échec de la connexion nous retournons la valeur 1. Le lecteur est invité à bien remarquer que nous n'utilisons pas une seule instruction return ou exit pour cela, mais la séquence SetExitCode(1); return;. Ceci est une particularité des applications Ultimate++.

2 CRÉATION DE TABLES

Nous abordons maintenant les choses sérieuses. La création de tables de données est le point le plus obscur, faute d'une bonne documentation. C'est dommage car l'approche d'Ultimate++ est originale et efficace. Nous découvrons son mécanisme par la pratique. Pour cela, créez une application de type 'Core console project', comme pour la connexion. Sur le CD-Rom, ou notre site, cet exemple est baptisé DemoCreateTable. Nous voulons créer une table décrivant les membres de l'équipe de Programmez!. Les colonnes seront: un id auto-incrémenté et clé primaire, le nom du membre, de type VARCHAR et indexé, le prénom du membre de type VARCHAR et le salaire, de type DOUBLE. Sous Ultimate++ une structure de table telle que celle-ci est décrite par un jeu de macros C++. Commencez par ajouter un fichier de nom database.sch à votre projet. Puis mettez-y ce contenu:

```
TABLE_(EQUIPE)
    SERIAL_ (EQUIPE_ID) PRIMARY_KEY
    STRING_ (EQUIPE_NOM, 50) INDEX
    STRING_ (EQUIPE_PRENOM, 50)
    DOUBLE_ (EQUIPE_SALAIRE)
END_TABLE
```

Tout ce code n'est que macros C++. Voyons où ces macros sont définies et comment elles sont utilisées. Pour cela, commençons par le contenu du fichier en-tête de notre application DemoCreateTable :

```
#ifndef _DemoCreateTable_DemoCreateTable_h
#define _DemoCreateTable_DemoCreateTable_h

#include <Core/Core.h>

using namespace Upp;

#include <PostgreSQL/PostgreSQL.h>
#define SCHEMADIALECT <PostgreSQL/PostgreSQLSchema.h>
#define MODEL <DemoCreateTable/database.sch>
#include "Sql/sch_header.h"

#endif
```

Ce contenu sera le même pour tous les autres exemples, nous n'y reviendrons plus. Par rapport à l'exemple de connexion, deux macros, SCHEMADIALECT et MODEL sont définies. La première macro s'étend sur le nom d'un en-tête du package PostgreSQL. La seconde s'étend sur le nom du fichier qui contient notre définition de table. Nous verrons le rôle de ces macros un peu plus loin. Vient ensuite l'inclusion de l'en-tête sch_header.h du package Sql. Toutes ces informations doivent obligatoirement être fournies par le programmeur tant qu'Ultimate++ n'est pas enrichi d'un assistant pour le faire. L'en-tête sch_header, comme son nom le suggère, contient des macros (encore...) qui servent d'en-tête au schéma que nous avons défini dans le fichier database.sch. Puis sch_header.h inclut le fichier défini par la macro SCHEMADIALECT. Si, curieux, vous consultez PostgreSQLSchema.h, vous y trouverez toutes les macros permettant de travailler avec les types compris par le système de base de données. Chaque système présentant ses particularités, les packages PostgreSQL, MySQL, Oracle, etc., offrent tous ces macros. Ainsi, petit miracle comme il y en a tant avec Ultimate++, il est possible de porter son code d'un SGDBR vers un autre seulement en modifiant la macro SCHEMADIALECT, ce qui est formidable. A son tour le fichier représenté par SCHEMADIALECT (pour nous PostgreSQLSchema.h) inclut le fichier Sql/sch_model.h qui à son tour inclut le fichier représenté par la macro MODEL, c'est-à-dire pour nous database.sch, c'est-à-dire encore le modèle de notre table. Quand ce mécanisme d'inclusion en chaîne se termine, une structure au sens C++ du terme est définie qui mappe la table que nous voulons créer. Voici maintenant le code de l'application:

```
#include <Core/Core.h>

#include "DemoCreateTable.h"

#include <Sql/sch_schema.h>
```

```
#include <Sql/sch_source.h>

CONSOLE_APP_MAIN
{
    PostgreSQLSession session;
    bool result;

    result = session.Open("host=localhost dbname=UppProgrammezDB
user=fred password=xxxx");
    session.SetTrace();
    if(!result)
    {
        Cout() << "Impossible d'ouvrir la connexion" << '\n';
        SetExitCode(1);
        return;
    }
    Cout() << "DemoCreatetable -- Connexion ouverte" << '\n';

    SqlSchema sch(PGSQL);
    StdStatementExecutor se(session);
    All_Tables(sch);

    if(sch.ScriptChanged(SqlSchema::UPGRADE))
        PostgreSQLPerformScript(sch.Upgrade(),se, 0);
    if(sch.ScriptChanged(SqlSchema::ATTRIBUTES)) {
        PostgreSQLPerformScript(sch.Attributes(),se, 0);
    }
    if(sch.ScriptChanged(SqlSchema::CONFIG)) {
        PostgreSQLPerformScript(sch.ConfigDrop(),se, 0);
        PostgreSQLPerformScript(sch.Config(),se, 0);
    }

    sch.SaveNormal();

    session.Close();
    Cout() << "Connexion fermee" << '\n';
}
```

Les deux fichiers en-têtes inclus dans ce code apportent les fonctionnalités pour générer des fichiers de requêtes SQL et travailler avec eux. Là encore cette inclusion est obligatoire et à la charge du programmeur. Regardons maintenant le code qu'il convient d'écrire pour aboutir à la création de la table. D'abord on instancie deux classes encapsulant la notion de schéma, et un "exécuteur de requête". Puis vient l'appel à la fonction `All_Tables`, globale à l'espace de nom `Upp`, définie dans `sql_schema.h`, et qui parcourt notre fichier `database.sch` pour générer le(s) script(s) SQL de création de table. On comprend ici que notre fichier `database.sch` pourrait très bien contenir la définition de plusieurs tables. Puis le code qui suit demande l'exécution des scripts SQL (ou pour être plus précis l'exécution du contenu des scripts) à condition que ceux-ci viennent d'être générés ou modifiés. Ainsi, si le programme d'exemple est lancé deux fois consécutivement, la table est créée au premier lancement et rien ne se passe au lancement suivant. En revanche, si nous modifions le schéma dans `database.sch`, par exemple en ajoutant une colonne à la table, l'exécution du programme provoquera l'ajout de la colonne dans la base de données. Il est important de

remarquer d'un autre côté que si nous effaçons la table de notre base de données au moyen d'un outil d'administration quelconque, relancer notre programme ne recréera pas la table! Ceci en raison de la présence des scripts SQL. Ces derniers doivent donc être effacés du disque dur dans une telle situation. Ils résident sous `upplout\` puis sous le répertoire dont le nom décrit à la fois le compilateur utilisé et le type d'application généré par Ultimate++ (Console ou GUI). Nous en avons terminé avec la partie la plus difficile. Et encore, elle n'est difficile que parce qu'elle n'est ni documentée ni automatisée.

3 INSÉRER DES DONNÉES DANS UNE TABLE

Regardons tout de suite le code de notre exemple `DemoInsert` :

```
#include "DemoInsert.h"

#include <Sql/sch_schema.h>
#include <Sql/sch_source.h>

CONSOLE_APP_MAIN
{
    PostgreSQLSession session;
    bool result;

    result = session.Open("host=localhost dbname=UppProgrammezDB
user=fred password=xxxx");
    session.SetTrace();
    if(!result)
    {
        Cout() << "Impossible d'ouvrir la connexion" << '\n';
        SetExitCode(1);
        return;
    }
    Cout() << "DemoInsert -- Connexion ouverte" << '\n';

    Sql sql(session);
    sql.ClearError();
    sql.Begin();
    sql * Delete(EQUIPE);
    sql * Insert(EQUIPE)
        (EQUIPE_NOM, "Kaminsky")
        (EQUIPE_PRENOM, "Jean")
        (EQUIPE_SALAIRE, "1000000.00")
    ;
    sql * Insert(EQUIPE)
        (EQUIPE_NOM, "Tonic")
        (EQUIPE_PRENOM, "Francois")
        (EQUIPE_SALAIRE, "10000.00")
    ;
    sql * Insert(EQUIPE)
        (EQUIPE_NOM, "Develay")
        (EQUIPE_PRENOM, "Sebastien")
        (EQUIPE_SALAIRE, "100.00")
    ;
    sql * Insert(EQUIPE)
        (EQUIPE_NOM, "Mazue")
        (EQUIPE_PRENOM, "Frederic")
    ;
}
```



```

(EQUIPE_SALAIRE, "1.00")
;

sql.Commit();

session.Close();
Cout() << "Connexion fermee" << '\n';
}

```

Ne nous arrêtons pas sur les frappantes et injustes disparités entre les salaires ;) La classe `Sql` encapsule la notion de curseur, et propose donc les méthodes `Begin` et `Commit` pour encadrer une transaction. Nous avons vu que les concepteurs d'Ultimate++ savent écrire des macros, nous découvrons maintenant qu'ils savent surcharger des opérateurs. Ainsi l'opérateur `*` permet, dans toutes les situations d'injecter les résultats d'une expression SQL dans un curseur. Pour bien tout comprendre regardons ce que nous trouvons avec et derrière `Insert` dans le code. `Insert` est une fonction inline définie dans l'en-tête `Sqlx.h` et qui retourne une instance de la classe `SqlInsert`. Celle-ci voit son opérateur `()` surchargé, ce qui permet cette syntaxe inédite qui serait rejetée par le compilateur sans cela. Finalement, grâce à la surcharge de deux opérateurs (`*` `()`) nous avons un code concis, expressif et facile à relire, et surtout à maintenir car il n'y pas besoin ni de manipuler du SQL en dur, ni de se soucier des particularités du SGDBR sous-jacent.

4 RÉCUPÉRER ET MODIFIER LES DONNÉES

Le code ci-dessous, extrait de l'exemple `DemoSelect` montre comment récupérer et modifier et afficher des données:

```

Sql sql(session);

sql.Execute("SELECT EQUIPE_NOM, EQUIPE_PRENOM, EQUIPE_SALAIRE
FROM EQUIPE");

while(sql.Fetch())
    Cout() << sql[0] << " " << sql[1] << " " << sql[2] << '\n';

Cout() << '\n' << "-----" << '\n';

sql * Select(EQUIPE_NOM).From(EQUIPE);
while(sql.Fetch())
    Cout() << sql[0] << '\n';

Cout() << '\n' << "-----" << '\n';

sql * Select(EQUIPE_NOM, EQUIPE_PRENOM).From(EQUIPE);
while(sql.Fetch())
    Cout() << sql[0] << " " << sql[1] << '\n';
    Cout() << '\n' << "---- Salaires inferieurs a 1000 ----" << '\n';

sql * Select(EQUIPE_NOM, EQUIPE_PRENOM).From(EQUIPE).Where(EQUIPE_
SALAIRE < 1000);
while(sql.Fetch())
    Cout() << sql[0] << " " << sql[1] << '\n';

```

```

Cout() << '\n' << "----- Petite augmentation -----" << '\n';

sql * Select(EQUIPE_NOM, EQUIPE_PRENOM).From(EQUIPE);
sql * Update(EQUIPE) (EQUIPE_SALAIRE, EQUIPE_SALAIRE +
10).Where(EQUIPE_SALAIRE > 1000);
sql * Select(EQUIPE_NOM, EQUIPE_PRENOM, EQUIPE_SALAIRE).From
(EQUIPE).OrderBy(EQUIPE_SALAIRE);
while(sql.Fetch())
    Cout() << sql[0] << " " << sql[1] << " " << sql[2] << '\n';

```

Nous retrouvons ici les surcharges des opérateurs `*` et `()`. On appréciera, grâce à ces surcharges, l'homogénéité du code: `Insert`, `Select` `Update` se comportent à l'identique. Nous rencontrons en outre une nouvelle surcharge, celle de l'opérateur `[]` qui permet de travailler avec le curseur SQL comme s'il s'agissait d'un simple tableau :)



Fig.4

5 UN EXEMPLE GRAPHIQUE

Celui-ci, `DemoGUI`, est une adaptation de l'unique exemple fourni avec Ultimate++.

Notre exemple `DemoGUI` en action.

[Fig.4] Nous renvoyons le lecteur au CD-Rom ou sur le site, et regardons seulement cet extrait :

```

void DemoGUI::Augmentation()
{

    Sql sql(session);
    sql * Select(EQUIPE_NOM, EQUIPE_PRENOM).From(EQUIPE);

    SqlUpdate usql(EQUIPE);

    usql(EQUIPE_SALAIRE, EQUIPE_SALAIRE + 10.0);
    usql.Where(EQUIPE_SALAIRE > 1000.0);

    sql * usql;
    // Pour le rafraîchissement
    // et l'ordre de l'affichage
    sqlarray.SetOrderBy(Descending(EQUIPE_SALAIRE));
    sqlarray.Query();
}

```

Nous y retrouvons des choses connues. Ce qui est intéressant est `sqlarray`, une instance de `SqlArray`. Cette classe affiche d'une part une grille à l'écran et d'autre part elle se comporte comme un curseur. C'est pourquoi `sqlarray.Query()` provoque le rafraîchissement de l'écran. Il est possible d'aller encore plus loin dans les relations données/composants graphiques.

Nous laissons cela pour une prochaine fois et espérons avoir donné au lecteur l'envie de découvrir plus avant ce formidable IDE qu'est Ultimate++.

■ Frédéric Mazué

fmaeze@programmez.com

Génération de documents en images

Différents formats de documents textes existent (.xps, .doc, .docx, .rtf, .pdf, .odf, etc.) et généralement ces formats sont propres aux applications pouvant les lire, ce qui ne facilite pas les choses d'un système d'exploitation à un autre, d'une version d'un lecteur à un autre (la rétrocompatibilité par exemple), etc. Le moyen d'éviter cela est de transformer ces fichiers en d'autres fichiers dans un format générique, pouvant être lu par toutes les plates-formes.

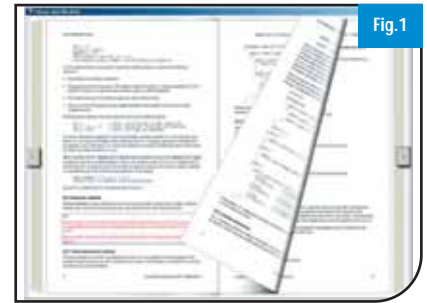
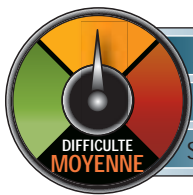


Fig.1



Fig.2



APPLICATION : DOCUMENT

LANGAGE : .NET

SOURCE : NON

Ces différents formats de documents sont composés de la même manière, ils comportent un certain nombre de pages pouvant contenir du texte, des images,

etc. Pouvoir transformer ces pages en fichiers images permet de les consulter plus facilement par la suite, sans avoir à passer par un lecteur propriétaire en particulier. Voici les deux méthodes qui peuvent être utilisées pour arriver à ce résultat.

GÉNÉRATION PAR DLL POUR LE FORMAT XPS

Le format XPS (XML Paper Specification) est un format de document déposé par Microsoft pour concurrencer le format PDF. Pour pouvoir lire un document XPS, il faut installer le Microsoft XPS Viewer qui utilise le Framework 3.0. Pour plus d'informations il est possible de consulter les spécifications de ce format sur le site de Microsoft (<http://www.microsoft.com/whdc/xps/xpsspec.mspx>).

Pour pallier l'utilisation de ce lecteur, il est possible de générer des images (au format .png pour que ce soit plus léger pour le disque dur) afin de pouvoir consulter ce document plus aisément.

Pour cela, il faut utiliser l'espace de noms System.Windows.Xps.Packaging et utiliser les méthodes suivantes :

```
public int SaveXpsPageToBitmap(string xpsFileName, string name)
{
    // XPS Document
    XpsDocument xpsDoc = new XpsDocument(xpsFileName, System.IO.
    FileAccess.Read);
    FixedDocumentSequence docSeq = xpsDoc.GetFixedDocument
    Sequence();

    // The number of pages
    PageCount = docSeq.References[0].GetDocument(false).Pages.
    Count;

    // Convert a XPS page to a PNG file
    for (int pageNum = 0; pageNum < PageCount; pageNum++)
    {
        DocumentPage docPage = docSeq.DocumentPaginator.GetPage
```

```
(pageNum);
    BitmapImage bitmap = new BitmapImage();
    RenderTargetBitmap renderTarget =
        new RenderTargetBitmap((int)docPage.Size.Width,
                                (int)docPage.Size.Height,
                                96, 96,
                                System.Windows.Media.Pixel
    Formats.Pbgra32);
    renderTarget.Render(docPage.Visual);

    PngBitmapEncoder encoder = new PngBitmapEncoder();
    encoder.Frames.Add(BitmapFrame.Create(renderTarget));

    FileStream pageOutputStream = new FileStream(name + ".Page" + page
    Num + ".png", FileMode.Create, FileAccess.Write);

    encoder.Save(pageOutputStream);
    pageOutputStream.Close();
    }
    return PageCount;
}
```

GÉNÉRATION AVEC DES OBJETS COM VIA PDFCREATOR POUR LES AUTRES FORMATS

Les autres formats de fichiers (.pdf, .doc, .docx, .rtf, .pdf, .odf) peuvent être générés sous forme d'images avec l'outil PDFCreator, pour cela il faut utiliser les objets COM de cette application. Cette solution est un peu moins intuitive et fait appel à un élément extérieur par rapport à celle utilisée pour les XPS, peu de documentation existe malheureusement pour manipuler au mieux ces objets COM. Voici le code utilisé afin de pouvoir traiter les documents du type .doc, .docx, .rtf, .pdf, .odf. Tout d'abord, l'activation d'une instance de l'objet COM de PDFCreator :

```
string parameters;
pErr = new PDFCreator.clsPDFCreatorError();
_PDFCreator = new PDFCreator.clsPDFCreator();
_PDFCreator.eError += new PDFCreator.__clsPDFCreator_eErrorEvent
```

```

Handler(_PDFCreator_eError);

_PDFCreator.eReady += new PDFCreator.__clsPDFCreator_eReady
EventHandler(_PDFCreator_eReady);

parameters = "/NoProcessingAtStartup";

// ouverture de l'objet COM PDFCreator
if (!_PDFCreator.cStart(parameters, false))
{
    System.Windows.MessageBox.Show("Attention", "PDFCreator
est déjà ouvert !");
    this.Close();
}

```

Ensuite, les options d'impression sont définies :

```

// option d'impression
opt.UseAutosave = 1;
opt.UseAutosaveDirectory = 1;
opt.AutosaveDirectory = fi.DirectoryName;
opt.AutosaveFormat = FileType;
opt.AutosaveStartStandardProgram = 1;
opt.PNGColorscount = 1;
opt.OptionsVisible = 0;
opt.NoProcessingAtStartup = 1;
opt.AutosaveFilename = fname;
_PDFCreator.cOptions = opt;

_PDFCreator.cClearCache();

```

Puis, l'imprimante PDFCreator est définie comme l'imprimante par défaut :

```

// définition de l'imprimante par défaut
DefaultPrinter = _PDFCreator.cDefaultPrinter;
PDFCreator.cDefaultPrinter = "PDFCreator";

```

Ensuite l'impression peut commencer :

```

// impression
_PDFCreator.cPrintFile(fi.FullName);

ReadyState = false;
_PDFCreator.cPrinterStop = false;

// attente jusqu'à la fin de l'impression
do
{
    System.Windows.Forms.Application.DoEvents();
} while (!ReadyState && _PDFCreator.cProgramIsRunning);

_PDFCreator.cPrinterStop = true;

// retour de l'imprimante par défaut du départ
_PDFCreator.cDefaultPrinter = DefaultPrinter;

```

Enfin la méthode qui permet de fermer l'objet COM, lorsque l'impression est terminée :

```

private void ClosePDFCreator()
{
    // fermeture de l'objet COM PDFCreator
    _PDFCreator.cClose();
    System.Runtime.InteropServices.Marshal.ReleaseComObject
(_PDFCreator);
    System.Runtime.InteropServices.Marshal.ReleaseComObject
(pErr);
    pErr = null;
    GC.Collect();
    GC.WaitForPendingFinalizers();
    GC.Collect();
    GC.WaitForPendingFinalizers();
}

```

Dans cet exemple, le format PNG est utilisé pour la génération d'images, qui permet une plus grande compression des images que le BMP et donc les fichiers générés tiennent moins de place sur le disque dur. Par exemple, pour un document de 111 pages, en les générant en BMP j'obtiens un dossier pesant 690 Mo et en PNG seulement 5 Mo !

LECTEURS D'IMAGES-DOCUMENTS

Voici 2 exemples de visionneuses d'images téléchargeables sur Internet. XPSReader (<http://www.codeplex.com/XPSReader>)

Basé sur la technologie WPF de Microsoft et à l'aide du control WPFBookControl de Mitsu Furuta (<http://www.codeplex.com/wpfbookcontrol>), ce projet permet de lire des documents XPS (XML Paper Specification) en générant des images des pages du document choisi préalablement. Les images générées sont au format PNG. [Fig.1]

Il est possible de tourner les pages à la souris pour donner un effet livre au document.

MANGA READER (<http://mangareader.wordpress.com>)

Egalement basé sur la technologie WPF, Manga Reader est un lecteur de mangas. Il prend en charge plusieurs formats d'images (JPEG, PNG, GIF, BMP, TIFF, WDP) [Fig.2]. Cette application propose plusieurs effets graphiques : il est ainsi possible de tourner les pages (comme dans le projet présenté ci-dessus) ou de zoomer librement en avant ou en arrière sur les bandes dessinées.

CONCLUSION

Pour conclure, le traitement avec des objets COM est un peu plus contraignant que la génération par DLL car il faut s'appuyer sur une application externe, en l'occurrence PDFCreator ici. La seule contrainte est que lors de la génération des images, il est impératif que le lecteur soit installé sur la machine permettant la transformation. Ensuite, les images peuvent être consultées sur d'autres machines contenant un simple visionneur d'images supportant le type d'images générées. Mais les deux résultats sont plutôt satisfaisants et permettent une lecture agréable des différents documents. Un lecteur Web peut être désormais envisagé grâce au control Silverlight développé par Mistu Furuta sur le modèle du control WPF déjà existant (<http://www.codeplex.com/wpfbookcontrol>).

■ Audrey PETIT

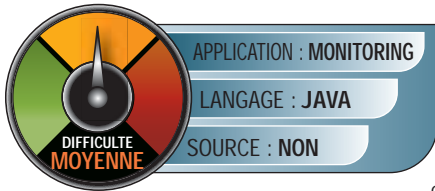
<http://blogs.developpeur.org/audrey>

Ingénieur Consultante pôle RIDA

(Rich Internet and Desktop Applications), Winwise

Health Monitoring : le journal à surveiller

Lorsque l'on développe une application Web, il est fréquent de vouloir loguer les différents événements survenant sur cette dernière. Par défaut, lorsqu'une erreur est détectée elle est automatiquement enregistrée dans le journal d'événements de Windows. Ce système est basé sur ce que l'on appelle « *Health Monitoring* ».



Ce principe permet de loguer les différents événements d'une application Web. En général on ne modifie que le fichier de configuration pour le prendre en compte. Par défaut, il est automatiquement activé dans le fichier de configuration global de la machine. Ce dernier se trouve dans C:\WINDOWS\Microsoft.NET\Framework\v2.0.XXXXX\CONFIG. Nous allons maintenant voir comment configurer ce système pour votre application.

FICHIER DE CONFIGURATION

Comme je l'ai dit précédemment, le système de log « *Health Monitoring* » est essentiellement basé sur les fichiers de configuration. Si vous n'avez pas de fichier de configuration, faites un clic droit sur votre projet puis « *Ajouter un nouvel élément...* » et choisissez « Fichier de configuration Web ». Par défaut, il contient déjà un certain nombre d'informations. Nous allons modifier le nœud *system.web* pour gérer notre système de logs selon le modèle suivant :

```
<configuration>
  <system.web>
    <healthMonitoring enabled=>true heartbeatInterval=>0>
      <bufferModes>
      </bufferModes>
      <providers>
      </providers>
      <eventMappings>
      </eventMappings>
      <profiles>
      </profiles>
      <rules>
      </rules>
    </healthMonitoring>
  </system.web>
</configuration>
```

LES DÉCLENCHEURS (EVENTMAPPING)

Les déclencheurs permettent de spécifier quel type d'événement peut déclencher l'action de log. Afin de mieux comprendre ce mécanisme voici les différents événements pouvant survenir : [Fig.1]

- **WebHeartbeatEvent** : Donne des informations sur l'état de l'application. On peut ainsi être informé sur le nombre de threads actifs, les requêtes, ... Il est déclenché de façon périodique. Pour indiquer l'intervalle de temps entre chaque déclenchement, il faut modifier l'attribut *heartbeatInterval* du nœud *healthMonitoring*.
- **WebRequestEvent** : Contient toutes les informations sur les

requêtes exécutées vers l'application. On peut ainsi trouver toutes les requêtes passées.

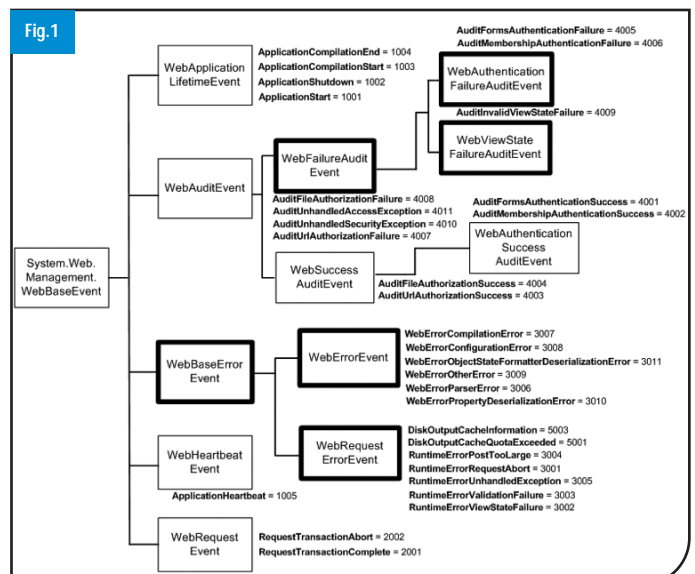
- **WebAuditEvent** : Donne les informations relatives à la sécurité. On trouvera, entre autres, des informations sur les authentifications réussies ou échouées des *Membership*.
- **WebBaseErrorEvent** : Classe de base qui contient toutes les événements relatifs aux erreurs. On trouvera aussi bien les erreurs de compilation que les erreurs de configuration ou alors les exceptions non gérées...
- **WebApplicationLifeTimeEvent** : Contient des informations sur le cycle de vie de l'application telles que le démarrage, l'arrêt ou la compilation de l'application.

Chaque événement contient un code (*eventCode*) permettant de l'identifier. Dans le schéma précédent vous pouvez apercevoir les principaux. Il est ainsi possible de ne traiter que certains événements en fonction de leur code. Pour ajouter un déclencheur, il faut remplir la section *eventMappings* :

```
<eventMappings>
  <clear/>
  <add name=>All Errors>
    type=>System.Web.Management.WebBaseErrorEvent>
    startEventCode=>0>
    endEventCode=>2147483647> />
</eventMappings>
```

Il faut donc renseigner les attributs suivants :

- **name** : Libellé de l'événement que l'on utilisera dans le reste du fichier de configuration.
- **type** : Type de l'événement.
- **startEventCode** : Code à partir duquel on commence à traiter les événements.



- `endEventCode` : Code à partir duquel on arrête de traiter les événements.

La balise `clear` permet de supprimer les informations contenues dans cette section mais dans un autre fichier de configuration. Par exemple celui de l'ordinateur.

LES FOURNISSEURS (PROVIDERS)

Lorsqu'un événement survient, il faut ensuite le traiter, c'est-à-dire l'enregistrer dans une base de données, l'envoyer par mail, ... C'est le rôle du fournisseur. Il en existe différents : [Fig.2]

SIMPLEMAILWEBEVENTPROVIDER

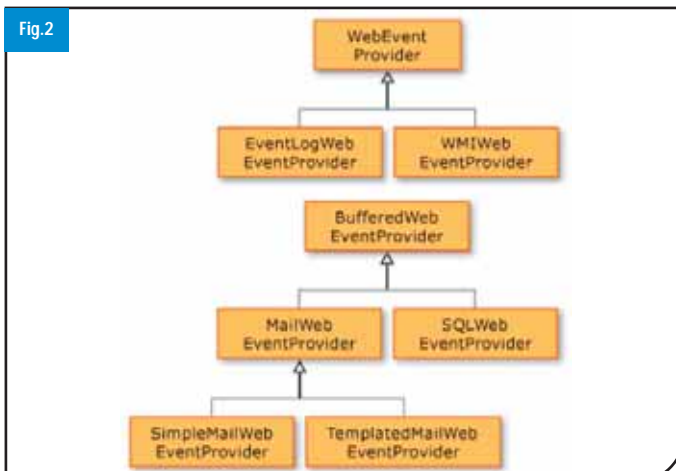
Ce fournisseur va permettre l'envoi de mails. Ce dernier contiendra toutes les informations de l'événement. Dans un premier temps il est nécessaire de configurer le serveur SMTP afin que l'application sache comment envoyer un message. Si cette étape n'est pas faite, aucune erreur ne sera générée mais aucun mail ne sera envoyé. Pour configurer cette information, il faut modifier le fichier de configuration comme ceci :

```
<system.net>
  <mailSettings>
    <smtp>
      <network host=>HostAddress username=>UserName password=>Password
      port=>Port />
    </smtp>
  </mailSettings>
</system.net>
```

Il faut ensuite remplir la section `providers` :

```
<providers>
  <clear/>
  <add name=>CriticalMailEventProvider
    type=>System.Web.Management.SimpleMailWebEventProvider
    from=>Email
    to=>Email
    cc=>Email
    bcc=>Email
    bodyHeader=>Attention !>
    bodyFooter=>Merci de corriger cette erreur au plus vite.>
    subjectPrefix=>[ApplicationName][Error] />
  </providers>
```

Fig.2



Voici les différents attributs :

- `name` : Nom qui permettra d'identifier ce dernier dans le reste du fichier de configuration.
- `type` : Type du fournisseur.
- `from` : Adresse qui envoie le message.
- `to` : Adresse du destinataire.
- `cc` : Adresse en copie.
- `bcc` : Adresse en copie cachée.
- `bodyHeader` : Texte à afficher dans l'entête du message.
- `bodyFooter` : Texte à afficher dans le pied du message.
- `subjectPrefix` : Préfixe pour le sujet du message. En général on indique le nom de l'application ainsi que le type du message. Cela simplifie la gestion des mails.

TEMPLATEDMAILWEBEVENTPROVIDER

Ce fournisseur est presque identique au précédent, sauf que lui va se baser sur un modèle.

```
<providers>
  <clear/>
  <add name=>CriticalMailEventProvider
    type=>System.Web.Management.TemplatedMailWebEventProvider
    from=>Email
    to=>Email
    cc=>Email
    bcc=>Email
    subjectPrefix=>[ApplicationName][Error]>
    template=>~/Formatter.aspx>
    detailedTemplateErrors=>true>
  />
</providers>
```

- `template` : Page web qui va formater le mail au format HTML.
- `detailedTemplateErrors` : Permet d'afficher les erreurs survenues dans la page template. Cette option est très utile pour le débogage mais doit être supprimée lors de la mise en production.

Voyons un peu plus en détail la page `Formatter.aspx` :

```
<body>
  <form id=>form1 runat=>server>
  <div>
    <asp:GridView ID=>gridView runat=>server
      AutoGenerateColumns=>true>
      EnableViewState=>>false>
    </asp:GridView>
  </div>
</form>
</body>
```

Dans le code de cette page nous avons juste ajouté un GridView. Maintenant regardons la page `Formatter.aspx.cs` :

```
protected void Page_Load(object sender, EventArgs e)
{
  if (TemplatedMailWebEventProvider.CurrentNotification != null)
  {
    gridView.DataSource =
      TemplatedMailWebEventProvider.CurrentNotification.Events;
```

```
gridView.DataBind();
}
}
```

Dans le code behind, nous associons tous les évènements survenus comme source de données du GridView. Le contenu de la page sera ensuite envoyé par mail en cas d'erreur survenant sur l'application. Bien entendu, il est possible de faire des pages beaucoup plus évoluées.

SQLWEBEVENTPROVIDER

Ce fournisseur permet d'enregistrer toutes les informations en base de données. Pour cela vous devez créer les tables nécessaires dans la base de données en utilisant l'outil aspnet_regsql :

- Authentification Windows :

```
aspnet_regsql.exe -E -S serverName -d database -A w
```

- Authentification SQL Server :

```
aspnet_regsql.exe -U userID -P password -S serverName -d database -A w
```

Voyons maintenant le fichier de configuration :

```
<providers>
  <clear />
  <add name=>SqlWebEventProvider>
    type=>System.Web.Management.SqlWebEventProvider>
    connectionStringName=>LogCnx>
    maxEventDetailsLength=>1073741823>
</providers>
```

- name : Nom du fournisseur permettant de l'identifier dans le reste du fichier de configuration.
- type : Type du fournisseur.
- connectionStringName : Nom de la chaîne de connexion qui doit être renseignée dans la section connectionStrings du fichier de configuration.
- maxEventDetailsLength : Nombre de caractères maximal pour enregistrer l'erreur. Il correspond à la taille de la colonne Details de la table aspnet_WebEvent_Events dans la base de données, soit ici la taille d'un ntext.

EVENTLOGWEBEVENTPROVIDER

Ce fournisseur permet de logger les erreurs dans le journal d'évènements de Windows. Il n'est pas possible de spécifier la catégorie de l'évènement à moins de créer son propre fournisseur. Voici le fichier de configuration associé :

```
<providers>
  <clear />
  <add name=>EventLogProvider>
    type=>System.Web.Management.EventLogWebEventProvider> />
</providers>
```

TRACEWEBEVENTPROVIDER

Ce fournisseur enregistre les évènements dans les messages de trace ASP.NET. Voici son fichier de configuration :

```
<providers>
  <clear />
```

```
<add name=>TraceEventProvider>
  type=>System.Web.Management.TraceWebEventProvider> />
</providers>
```

WMIWEBEVENTPROVIDER

Ce fournisseur lie les évènements ASP.NET aux évènements de Windows Management Instrumentation (WMI). Voici le fichier de configuration associé :

```
<providers>
  <add name=>WmiWebEventProvider>
    type=>System.Web.Management.WmiWebEventProvider> />
</providers>
```

Ce dernier étant assez complexe, nous ne le traiterons pas en détail dans cet article. Vous trouverez plus d'informations sur le site de la MSDN : <http://msdn.microsoft.com/en-us/library/ms178713.aspx>

LE BUFFER

La mise en place d'un buffer est possible afin d'éviter la surcharge du serveur de mails ou de base de données. Un buffer permet par exemple de regrouper différents évènements et de les envoyer à intervalle de temps régulier. C'est pour cela qu'il ne fonctionne qu'avec les fournisseurs suivants :

- SqlWebEventProvider
- SimpleMailWebEventProvider
- TemplatedMailWebEventProvider

Pour l'activer, il suffit de rajouter les attributs suivants dans le nœud du fournisseur :

```
buffer=>true>
bufferMode=>Analysis>
```

L'attribut buffer permet d'activer ce dernier. Le bufferMode quant à lui permet de spécifier le nom du buffer que l'on désire utiliser. C'est pour cela que nous allons rajouter un nœud BufferMode dans notre fichier de configuration.

```
<healthMonitoring enabled=>true> heartbeatInterval=>0>>
  <!--...-->
  <bufferModes>
    <clear/>
    <add name=>Analysis>
      maxBufferSize=>100>
      maxFlushSize=>20>
      urgentFlushThreshold=>1>
      regularFlushInterval=>00:05:00>
      urgentFlushInterval=>00:01:00>
      maxBufferThreads=>1>
    />
  </bufferModes>
  <!--...-->
</healthMonitoring>
```

Voyons ensemble les différents attributs :

- Name : Libellé du buffer qui servira à l'identifier dans le fichier de configuration
- maxBufferSize : Valeur de type Int32 qui permet de spécifier le

nombre d'événements qui peuvent être dans le buffer.

- maxFlushSize : Valeur de type Int32 indiquant le nombre maximum d'événements par envoi de données.
- urgentFlushThreshold : Valeur de type Int32 qui permet de spécifier le seuil minimum pour l'envoi de données.
- regularFlushInterval : Spécifie l'intervalle de temps entre chaque envoi de données. Cette valeur est de type Int32.
- urgentFlushInterval : Spécifie l'intervalle de temps minimal entre chaque envoi de données. Cette valeur est de type TimeSpan.
- maxBufferThreads : Indique le nombre maximum de thread pour l'envoi de données. Cette valeur est de type Int32.

LES PROFILS (PROFILES)

Les profils permettent de gérer plus finement le déclenchement des événements.

```
<profiles>
  <clear/>
  <add name=>Default minInstances=>1 maxLimit=>Infinite
    minInterval=>00:10:00/>
</profiles>
```

- name : Nom permettant d'identifier le profil.
- minInstances : Nombre d'événements minimum avant de loguer.
- maxLimit : Nombre maximum d'événements devant être logués.
- minInterval : Temps minimum avant de loguer deux événements identiques.

LES RÈGLES (RULES)

Maintenant que nous avons défini les fournisseurs ainsi que les événements, il est nécessaire d'associer chaque événement à un ou plusieurs fournisseurs, c'est ce que l'on appelle les règles.

```
<rules>
  <clear />
  <add name=>All Errors Default
    eventName=>All Errors
    provider=>SqlWebEventProvider
    profile=>Default />
</rules>
```

- name : Nom de la règle.
- eventName : Nom de l'événement associé à cette règle. Il doit exister dans la section eventMappings.
- provider : Nom du fournisseur associé à cette règle. Il doit exister dans la section providers.
- profile : Nom du profil associé à cette règle. Il doit exister dans la section profile.

EXEMPLE DE FICHIER DE CONFIGURATION

Voici un fichier de configuration complet. Il envoie un mail dès qu'une erreur survient dans l'application. Le fournisseur vide les informations d'événements toutes les 5 minutes. Il vide jusqu'à 100 événements par notification et met en mémoire tampon jusqu'à 1000 événements en cas d'une augmentation soudaine de la fréquence des événements. Le fournisseur garantit que la fréquence d'envoi des événements ne sera pas supérieure à une fois par minute.

```
<healthMonitoring enabled=>true heartbeatInterval=>0>
  <bufferModes>
    <add name=>Analysis
      maxBufferSize=>1000
      maxFlushSize=>100
      urgentFlushThreshold=>100
      regularFlushInterval=>00:05:00
      urgentFlushInterval=>00:01:00
      maxBufferThreads=>1 />
  </bufferModes>
  <eventMappings>
    <clear/>
    <add name=>All Errors
      type=>System.Web.Management.WebBaseErrorEvent
      startEventCode=>0
      endEventCode=>2147483647 />
  </eventMappings>
  <providers>
    <clear/>
    <add name=>CriticalMailEventProvider
      type=>System.Web.Management.SimpleMailWebEventProvider
      from=>noreply@itelios.com
      to=>error@itelios.com
      buffer=>true
      bufferMode=>Analysis
      bodyHeader=>Attention !
      bodyFooter=>Merci de corriger cette erreur au plus vite.
      subjectPrefix=>[Programmez][Error] />
  </providers>
  <profiles>
    <clear/>
    <add name=>Default minInstances=>1 maxLimit=>Infinite
      minInterval=>00:10:00/>
  </profiles>
  <rules>
    <clear />
    <add name=>All Errors Default
      eventName=>All Errors
      provider=>CriticalMailEventProvider
      profile=>Default />
  </rules>
</healthMonitoring>
```

CONCLUSION

On a ainsi pu voir à travers cet article que la mise en place de logs dans ASP.NET est extrêmement simple grâce au Health Monitoring. Afin d'avoir une gestion encore plus fine des logs, il ne faut pas hésiter à combiner les différents fournisseurs et événements. Par exemple, il peut être utile de n'envoyer par mail que les erreurs critiques mais par contre d'enregistrer en base de données tous les événements. Dans certains cas, il est possible que les fournisseurs donnés par défaut ne vous suffisent pas, sachez que vous pouvez définir votre propre fournisseur. Pour plus d'informations référez-vous au site de la MSDN :

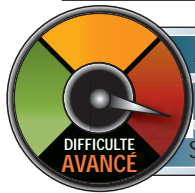
<http://msdn.microsoft.com/en-us/library/ms227676.aspx>.



■ Xavier Millot – Ingénieur d'étude et de développement – Itelios – <http://www.itelios.com>

Utilisation et exploitation des expressions régulières en .NET

Cet article vous propose une description détaillée des expressions régulières et des possibilités qu'elles vous offrent dans le cadre de vos applications .net.



APPLICATION : Expression régulière

LANGAGE : .net

SOURCE : non

Très utilisées lorsqu'il s'agit d'effectuer des traitements sur les chaînes de caractères, les

expressions régulières sont souvent assez mal connues des développeurs qui n'en utilisent pas toute la puissance. Dans cet article nous allons faire un tour des possibilités offertes par les *regex*, que ce soit pour faire des recherches dans du texte ou bien pour y effectuer des modifications.

Syntaxe des expressions régulières

Les expressions régulières permettent de rechercher des motifs (patterns) dans des chaînes de caractères, et d'effectuer des traitements dessus : extraction d'un certain motif dans une chaîne, remplacement ou découpage d'une chaîne. La syntaxe des expressions régulières permet d'effectuer des traitements très avancés sur les chaînes de caractères. Découvrons là :

Les littéraux

Les valeurs littérales représentent exactement une chaîne de caractères. Il peut s'agir de n'importe quel caractère n'étant pas un méta-caractère, ou bien un méta-caractère précédé d'un \

Le motif **"Exemple\."** permet de rechercher la sous-chaîne exacte "Exemple." dans une chaîne de caractère.

Les méta-caractères

Ce sont des caractères spéciaux qui ont une signification particulière lors de l'analyse de la regex. Ces caractères sont les suivants :

\	Caractère d'échappement Attention : en C#, le \ est déjà le caractère d'échappement dans les chaînes de caractères. Pour pouvoir l'utiliser il faudra le doubler, ou bien faire précéder votre chaîne de caractères du caractère '@'.
.	N'importe quel caractère, sauf retour à la ligne (\n) (si le mode ligne simple est actif, correspond à n'importe quel caractère, y compris le \n)
^	Début de chaîne (ou de ligne si en mode multiligne)
\$	Fin de chaîne (ou de ligne si en mode multiligne)
	Alternative
[]	Classe de caractères
()	Regroupements/captures/assertions
* ? + { }	Quantificateurs

Les quantificateurs

Ces méta-caractères s'appliquent au littéral ou au groupe le précédant. Ils définissent le nombre de fois où ce littéral ou groupe doit se répéter.

*	0, 1 ou plusieurs répétitions
?	0 ou 1 répétition
+	1 ou plusieurs répétitions
{x}	x est un nombre entier exactement x répétitions
{x,}	x est un nombre entier au moins x répétitions
{x,y}	x et y sont des nombres entiers x au minimum, y au maximum

Ex :

"a+" représente le caractère 'a' présent au moins une fois. Il valide les chaînes du type : " a ", " aa ", " aaaaaa ", etc.

"ab*a" valide les chaînes " aa ", " aba ", " abba ", " abbbbbbbba ", etc.

"ab{1,3}a" valide uniquement les chaînes " aba ", " abba " et " abbba "

Les alternatives

Les alternatives permettent de chercher une expression qui peut avoir plusieurs formats différents. On utilise le méta-caractère | pour séparer les différents formats possibles.

Ex : **"ch(ien|at)"** représente les caractères " ch " suivis soit des caractères " ien ", soit des caractères " at ". Donc :

Regex.Matches("j'ai un chien et un chat", "ch(ien|at)");

Retournera une collection avec les correspondances du mot " chien " et du mot " chat ". (Nous verrons plus loin en détail le fonctionnement de cette méthode)

Les classes de caractères

Une classe de caractères correspond à une liste de caractères sur lesquels on peut effectuer une correspondance. Il existe une liste de classes prédéfinies, et il est possible de définir ses propres classes de caractères selon ses besoins.

Définir une classe

Pour définir une classe de caractères, il faut écrire entre crochets la liste des caractères sur lesquels on veut effectuer la correspondance.

Ex : **"[abc]"** représente un des 3 caractères : 'a', 'b' ou 'c'.

Les méta-caractères sont considérés comme des littéraux dans les classes, il n'y a pas besoin de les échapper. Les caractères spéciaux dans une classe de caractères sont :

^ placé juste après le crochet ouvrant, il définit une classe complétée : les caractères acceptés sont ceux qui ne font pas partie de la liste. Placé ailleurs, il se comporte comme un littéral.

Ex : **"[^a]"** représente n'importe quel caractère différent de 'a'

- représente un intervalle.

Ex : **"[0-9]"** représente tous les chiffres de 0 à 9. Placé en début ou en fin de classe, il est considéré comme un littéral.

] Le crochet fermant représente la fin de la classe. Si on veut qu'il

soit reconnu comme littéral, il doit être placé en première position (une classe vide n'est pas reconnue comme une classe valide).

Ex : `[]` ne représente pas 2 classes vides, mais bien une classe contenant les caractères `]` et `[`.

`Regex.Matches("Oui, j'utilise des regex !", "[^A-Za-z]");`

retournera une collection avec chacun des caractères n'étant pas une lettre (majuscule ou minuscule) : la virgule, l'apostrophe, le point d'exclamation et chacun des espaces.

Nous avons vu que les caractères spéciaux dans une classe peuvent être considérés comme des littéraux selon leur position. Ainsi, même ces caractères spéciaux n'ont pas besoin d'être échappés pour être utilisés dans une classe.

Ex : `[]^` représente un des 3 caractères : `]`, `^` ou `['`.

Il commence par un crochet ouvrant et se termine par un crochet fermant, nous avons bien une classe de caractères.

Le premier caractère de la classe : `]`

Placé en première position, il se comporte comme un littéral. (Cela aurait aussi fonctionné dans une classe complémentée, le crochet doit alors être positionné juste après le `^`, il reste le premier littéral de la classe)

Le second caractère : `^`

Il n'est pas positionné juste après le crochet ouvrant, il ne sert donc pas à écrire une classe complémentée, mais représente le caractère `^`

Le troisième caractère : `-` Il n'est pas positionné entre deux autres caractères, il représente le caractère `-`

Les classes prédéfinies

.	Représente n'importe quel caractère sauf <code>\n</code> (ou n'importe quel caractère y compris le <code>\n</code> , si la regex est en mode ligne simple)
<code>\p{nom}</code>	Représente les caractères d'une catégorie Unicode spécifiée par son nom
<code>\P{nom}</code>	Représente les caractères n'appartenant pas à la catégorie Unicode spécifiée
<code>\w</code>	Représente n'importe quel caractère alphanumérique
<code>\W</code>	Représente n'importe quel caractère non-alphanumérique
<code>\s</code>	Représente un caractère d'espacement (espace, retour à la ligne, tabulation...)
<code>\S</code>	Représente un caractère n'étant pas un caractère d'espacement
<code>\d</code>	Représente un chiffre. Equivalent de <code>[0-9]</code>
<code>\D</code>	Représente un caractère qui n'est pas un chiffre. Equivalent de <code>[^0-9]</code>

Les captures

Les parenthèses permettent de capturer des sous-expressions dans le motif de l'expression régulière. Les captures sont automatiquement numérotées : la première (numéro 0) correspond à l'ensemble de la chaîne capturée par le motif ; les suivantes sont numérotées dans l'ordre des parenthèses ouvrantes.

Ex : Le motif `"#[0-9+]#"` fera une recherche d'une suite d'un ou plusieurs nombres, précédée et suivie du caractère `#`, et effectue la capture de la suite de nombres sans les `#`.

Il est possible de nommer les captures, afin de pouvoir les retrouver non pas par leur numéro mais par leur nom. On nomme une capture de la manière suivante :

`(?<nom>motif)` En commençant la capture par un point d'interrogation suivi du nom entre les caractères `'<'` et `'>'`, on a pu nommer la capture.

Il peut arriver qu'on veuille utiliser des parenthèses pour effectuer un groupement, sans pour autant vouloir capturer le groupement. On peut alors utiliser `"?: "` au début du groupement pour empêcher la capture de s'effectuer.

Ex : On peut utiliser des parenthèses pour délimiter plusieurs alter-

natives, sans pour autant vouloir les capturer. `"(ch(?:ien|at))"` Capturera les mots "chien" et "chat", mais pas "ien" ni "at".

Les références arrière

Les captures peuvent être utilisées pour vérifier qu'une certaine chaîne de caractères est présente en plusieurs endroits. Pour cela on utilise les références arrière : une référence arrière se note `"\n"`, où `n` correspond au numéro de la capture qu'on recherche. Attention : ceci ne fonctionne qu'avec des sous-chaînes déjà capturées lorsque le pointeur arrive à la référence, on ne peut pas référencer une capture située plus loin dans la chaîne.

Ex : si on veut faire une recherche sur des balises XML, on va chercher le même texte dans la balise ouvrante et la balise fermante. On peut utiliser un motif de ce style :

`"<(\w+)>(.*?)</\1>"`

Ici, le nom de la balise est capturé dans le premier groupe de parenthèses, et le `\1` dans la balise fermante permet de vérifier que le nom correspond bien à la balise ouvrante.

Les assertions

On peut considérer les assertions comme des conditions à vérifier pour valider le motif. Elles n'avancent pas la recherche dans la chaîne : elles ne servent qu'à vérifier une condition et non à récupérer ou remplacer une sous-expression.

Assertions automatiques

Nous avons déjà vu deux méta-caractères qui sont des assertions : `^` et `$`. Ces méta-caractères représentent le début et la fin de la chaîne, mais ne sont pas des caractères en tant que tels. Le motif `"^a"` représente uniquement le caractère "a", à condition qu'il soit placé en début de chaîne.

Vous pouvez utiliser les assertions suivantes :

<code>^</code>	La correspondance doit se trouver au début de la chaîne de caractères (ou en début de ligne si la regex est en mode multiligne)
<code>\$</code>	La correspondance doit se trouver à la fin de la chaîne de caractère (ou en fin de ligne, si la regex est en mode multiligne)
<code>^A</code>	La correspondance doit se trouver au début de la chaîne de caractères (ignore l'option multiligne)
<code>^Z</code>	La correspondance doit se trouver à la fin de la chaîne de caractères, ou avant un <code>\n</code> situé à la fin de la chaîne de caractères (ignore l'option multiligne)
<code>^z</code>	La correspondance doit se trouver à la fin de la chaîne de caractères (ignore l'option multiligne)
<code>\G</code>	La correspondance doit se trouver au point où s'est terminée la correspondance précédente. Ceci assure que les correspondances se suivent lorsque vous utilisez <code>Match.NextMatch()</code>
<code>\b</code>	La correspondance doit se trouver entre un caractère alphanumérique (<code>\w</code>) et un caractère non-alphanumérique (<code>\W</code>). La correspondance se fait donc au début ou à la fin d'un mot.
<code>\B</code>	Assertion inverse de l'assertion <code>\b</code> : La correspondance doit se trouver entre 2 caractères alphanumériques ou entre 2 caractères non-alphanumériques.

Assertions " complexes "

Il est possible de créer quatre types d'assertions. Elles s'écrivent de la manière suivante :

`(?=motif)` Assertion avant positive

`(?!motif)` Assertion avant négative

`(?<=motif)` Assertion arrière positive

`(?<!motif)` Assertion arrière négative

Le motif peut être écrit comme n'importe quel autre motif d'une expression régulière. Il existe donc les assertions avant et arrière,

et les assertions positives et négatives. Voyons les différences :

Assertion positive / négative

Une assertion positive vérifie que le motif est bien présent. La différence avec l'utilisation directe du motif est que le pointeur de recherche dans la sous-chaine n'avance pas, rien n'est capturé par la regex à travers cette assertion. L'assertion négative vérifie quant à elle l'absence du motif : si le motif est présent à cet endroit, alors la regex ne trouvera pas de correspondance.

Assertion avant / arrière

La différence entre une assertion avant et une assertion arrière est l'endroit où est cherché le motif par rapport au pointeur. Une assertion avant recherchera le motif à la suite du pointeur, sur la partie de la chaîne qui n'a pas encore été analysée. Une assertion arrière recherche le motif sur la partie précédant le pointeur, qui vient donc d'être analysée.

Ex : Voyons la différence avec la chaîne " abcd " et les motifs suivants : "ab(?=ab)cd" et "ab(?!=ab)cd".

Le motif "ab(?=ab)cd" ne trouvera pas de correspondance, car elle fait une assertion avant : à l'endroit de l'assertion les caractères " ab " ont déjà été passés.

Le motif "ab(?!=ab)cd" lui trouvera la correspondance, car elle cherche les motifs avant le pointeur, et donc trouve les caractères " ab ".

Utilisation des regex en .net

La classe de base pour l'utilisation des expressions régulières est la classe `Regex`. Elle se trouve dans le namespace `System.Text.RegularExpressions`.

Validation d'une chaîne de caractères

La classe `Regex` propose plusieurs méthodes pour valider une chaîne de caractères. Un exemple typique : on veut vérifier si une chaîne de caractère correspond à une adresse e-mail. Pour cela, on va vérifier que l'adresse contient bien un @, et qu'elle ne contient que des caractères autorisés. On crée donc notre regex de la manière suivante :

```
Regex verifMail = new Regex(@"([a-z0-9._-]+@[a-z0-9._-]+\.[a-z]{2,6})");
String adresse = "Mon adresse est : guillaume.lacasa@bewise.fr";
```

Regex.IsMatch(String input) : retourne une valeur booléenne égale à **true** si *input* contient le motif de la regex, **false** sinon.

verifMail.IsMatch(adresse) ; renvoie donc **true**

Regex.Match(String input) : retourne une valeur de type **Match**. Si le motif a été trouvé dans la chaîne de caractère *input*, le **Match** correspond à sa première occurrence et sa propriété **Success** vaut **true**. Sinon sa propriété **Success** vaut **false**. La méthode **Match.NextMatch()** permet de récupérer les occurrences suivantes. **Regex.Matches(String input)** : retourne une valeur de type **MatchCollection** contenant toutes les occurrences du motif trouvées dans *input*. Ces trois méthodes peuvent être appelées avec un second paramètre de type **int**, pour spécifier à partir de quelle position effectuer la recherche.

Substitutions

Les expressions régulières permettent aussi d'effectuer des substitutions dans les chaînes de caractères. Pour cela, on utilise la méthode : **Regex.Replace(String input, String replacement)** : cette méthode effectue la recherche du motif dans la chaîne *input* et la remplace par la chaîne *replacement*. Il est possible de remplacer dans le résultat les sous-expressions qui ont été capturées. Pour cela, il

suffit de mettre $\$n$ (où n correspond au numéro de la capture) dans la chaîne *replacement*, afin de mettre le contenu de la capture correspondante. Ex : Si en html on veut repérer une adresse e-mail et créer automatiquement un lien :

```
Regex verifMail = new Regex(@"([a-zA-Z0-9._-]+@[a-zA-Z0-9._-]+\.[a-zA-Z]{2,6})");
String adresse = "Mon adresse est : guillaume.lacasa@bewise.fr";
String replaceString = "<a href=\"\$1\"> $1 </a>";
verifMail.Replace(adresse, replaceString);
```

Cet exemple va retourner la chaîne suivante :

Mon adresse est : guillaume.lacasa@bewise.fr

Découpage d'une chaîne

Les expressions régulières permettent aussi de découper des chaînes de caractères grâce à la méthode **Regex.Split(String input, String pattern)**. Le but de cette méthode est le même que pour la méthode **String.Split()**, la différence est dans le séparateur de la chaîne : au lieu d'utiliser des caractères ou des chaînes de caractères comme séparateurs, on peut utiliser des motifs complexes.

Ex : Si on veut avoir pour séparateur n'importe quelle suite de caractères non alphabétique, on peut utiliser la regex suivante :

```
Regex decoupeur = new Regex(@"^[^a-zA-Z]+");
decoupeur.Split("Ceci_est_une_liste_de_mots");
```

On récupérera un tableau contenant :

{"Ceci", "est", "une", "liste", "de", "mots"}

Options des expressions régulières

Le comportement de la classe **Regex** peut être adapté à l'aide de plusieurs options. Ces options sont définies dans l'énumération **RegexOptions** et sont définies soit lors de l'appel du constructeur de la classe **Regex**, soit en paramètre optionnel lors de l'utilisation des méthodes statiques de cette classe.

Voici les principales options utilisables :

IgnoreCase	Ne fait pas de différence entre majuscules et minuscules. (Attention, les caractères accentués ne sont par contre pas pris en compte. 'E' et 'e' seront considérés identiques, mais 'é' et 'e' sont différents)
Multiline	Mode multiligne. Les méta-caractères ^ et \$ ne représentent alors pas les débuts et fin de la chaîne complète, mais les débuts et fin de n'importe quelle ligne.
Singleline	Mode ligne simple. Le caractère point (.) correspond à tous les caractères, y compris le retour à la ligne (\n). Contrairement à ce qu'on pourrait croire, les modes Multiline et Singleline ne sont pas incompatibles : ils ne modifient pas les significations des mêmes méta-caractères.
RightToLeft	Inverse le sens de la recherche : elle se fait de droite à gauche au lieu de se faire de gauche à droite.
Compiled	Mode compilé. Ce mode accélère l'utilisation des regex, mais ralentit leur création. Il est conseillé d'utiliser ce mode lorsqu'une même regex est utilisée de nombreuses fois dans le programme afin d'améliorer les temps d'exécution.

CONCLUSION

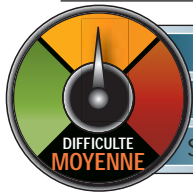
La puissance des expressions régulières, trop souvent méconnue, permet, lorsqu'elle est maîtrisée, de simplifier considérablement les traitements des chaînes de caractères.

■ Guillaume Lacasa

guillaume.lacasa@bewise.fr - Bewise

Optimisation de la bande passante et messaging temps réel avec BlazeDS

Une bonne RIA doit savoir relever deux défis: une interface riche et des échanges de données optimisés. En effet, dans le modèle web classique, nous rechargeons toute la page (et les données associées) lors d'un changement de vue. Avec une RIA, nous chargeons une fois les vues et l'application se concentre ensuite sur les échanges de données.



APPLICATION : RIA

LANGAGE : JAVA

SOURCE : NON

Ces derniers peuvent s'avérer gourmands et difficiles à contrôler. Je vais exposer dans cet article les différentes techniques de communication entre une application

Flex et un serveur d'application Java. Une application Flex peut évidemment communiquer avec des scripts PHP, Ruby, Perl, ASP.. mais c'est dans le monde Java que vous rencontrerez les mécanismes les plus évolués.

<mx:HTTPService>: de l'XML sur le réseau

L'appel le plus simple reste la requête HTTP qui vous renvoie un flux texte. Une servlet appelle un service Java, boucle sur les résultats pour construire un flux de données XML :

```
<%@page import="flex.samples.product.ProductService,
flex.samples.product.Product,
java.util.List"%>
<?xml version="1.0" encoding="utf-8"?>
<catalog>
<%
    ProductService srv = new ProductService();
    List list = null;
    list = srv.getProducts();
    Product product;
    for (int i=0; i<list.size(); i++)
    {
        product = (Product) list.get(i);
    }
    %>
    <product productId="<%= product.getProductID() %>">
        <name><%= product.getName() %></name>
        <description><%= product.getDescription() %></description>
        <price><%= product.getPrice() %></price>
        <image><%= product.getImage() %></image>
        <category><%= product.getCategory() %></category>
        <qtyInStock><%= product.getQtyInStock() %></qtyInStock>
    </product>
<%
    %>
</catalog>
```

Pour afficher dans une application Flex la liste des produits renvoyée par catalog.jsp, il faut procéder en trois étapes.

1. Déclarer le service avec <mx:HTTPService>
2. Associer le résultat de ce service à un composant DataGrid. La propriété dataProvider permet de pointer sur une collection de données.
3. Ajouter un bouton "Get Data" qui va lancer le service.

Voici le code Flex :

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml"
layout="vertical">
    <mx:HTTPService id="srv" url="http://localhost:8400/samples
/testdrive-http-service/catalog.jsp"/>

    <mx:DataGrid dataProvider="{srv.lastResult.catalog.product}"
width="100%" height="100%"/>

    <mx:Button label="Get Data" click="srv.send()"/>
</mx:Application>
```

Analyse des échanges de données sur le réseau

J'utilise alors le plug-in Firefox "Live HTTP Header" pour analyser le contenu du résultat HTTP et mesurer que pour un tableau de 18 lignes, 10Ko transitent sur le réseau. C'est du XML/texte brut, très verbeux. Ce mécanisme d'échange XML est standard dans la RIA avec les frameworks AJAX. Dans le cadre d'applications professionnelles avec des échanges de données volumineux, où la performance prime (dans le cadre d'une migration d'application client/serveur par exemple), cette réponse technologique ne suffit pas. Les développeurs recherchent alors des techniques de compression de la donnée, de paging, etc. Pour faire un appel à un Webservice qui renvoie un flux XML en réponse, utilisez directement le tag <mx:WebService> en indiquant l'url vers le wsdl.

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" backgroundColor="#FFFFFF">

    <mx:Script>
        <![CDATA[
            import mx.rpc.events.ResultEvent;
            private function handleResult(event:ResultEvent):void{
                dg.visible = true;
            }
        ]]>
    </mx:Script>
```

```

<mx:WebService id="srv" wsdl="http://www.livecycledata.org/
services/ProductWS?wsdl" showBusyCursor="true" result="hand
leResult(event)"/>

<mx:DataGrid editable="true" visible="false" dataProvider="
{srv.getProducts.lastResult}" width="353" height="100%" id="dg"/>

<mx:Button label="Get Data" click="srv.getProducts()"/>

</mx:Application>

```

C'est sensiblement le même code, à la différence près que dans cet exemple, je demande d'exécuter la fonction `handleResult` lorsque mon WebService me renvoie un résultat. Cette fonction rend visible le composant DataGrid. La gestion des appels de services est donc bien asynchrone, mais un WebService me renvoie toujours un flux XML verbeux et non typé. Je ne gagne donc rien en performances sur mon réseau.

<mx:RemoteObject>: du binaire sur le réseau

Dans le cadre de projets JEE, Adobe propose d'améliorer les échanges de données entre les applications Flex et les services Java à l'aide de la solution LiveCycle Data Services (un war à déployer sur le serveur de présentation). Une partie des Data Services est désormais gratuite et Open Source (LGPL) sous le nom de Blaze DS. Cette sortie s'accompagne de l'ouverture du format d'échange de données AMF (Action Message Format), du binaire qui décrit nativement les objets ActionScript 3 pour le lecteur Flash. Du coup, l'application client reçoit un flux compressé (gains de plus de 70% sur la bande passante par rapport au XML) et les données sont typées. Pour profiter du format AMF, il faut pouvoir appeler directement un service Java à l'aide du tag `<mx:RemoteObject>`. Voici un exemple de service Java avec une méthode `getProducts()` qui effectue une requête SQL pour constituer une liste d'objets.

```

import java.util.ArrayList;
import java.util.List;
import java.sql.*;
import flex.samples.ConnectionHelper;
import flex.samples.DAOException;

public class ProductService {
    public List getProducts() throws DAOException {
        List list = new ArrayList();
        Connection c = null;
        try {

```

```

        c = ConnectionHelper.getConnection();
        Statement s = c.createStatement();
        ResultSet rs = s.executeQuery("SELECT * FROM product ORDER BY name");
        while (rs.next()) {
            list.add(new Product(rs.getInt("product_id"),
                rs.getString("name"),
                rs.getString("description"),
                rs.getString("image"),
                rs.getString("category"),
                rs.getDouble("price"),
                rs.getInt("qty_in_stock")));
        }
    } catch (SQLException e) {
        e.printStackTrace();
        throw new DAOException(e);
    } finally {
        ConnectionHelper.close(c);
    }
    return list;
}

```

Blaze DS est une web app à encapsuler dans votre projet JEE. Il faut indiquer dans des fichiers de configuration XML sur le serveur, quels sont les services Java que l'on souhaite exposer aux applications Flex. Ce sont des destinations définies dans le fichier `remoting-config.xml` du répertoire `WEB-INF/flex`. Voici un extrait qui concerne notre service Java :

```

<default-channels>
    <channel ref="my-amf"/>
</default-channels>
<destination id="product">
    <properties>
        <source>flex.samples.product.ProductService</source>
    </properties>
</destination>

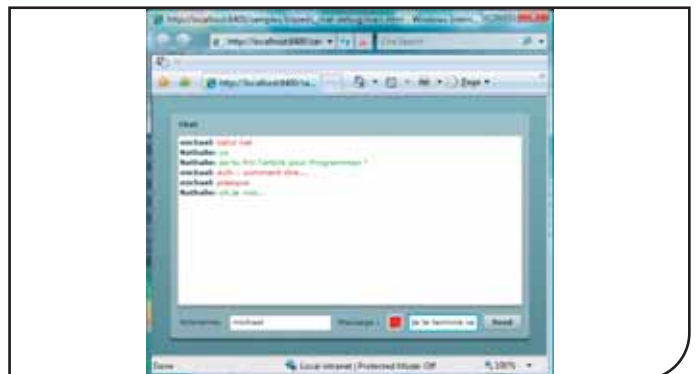
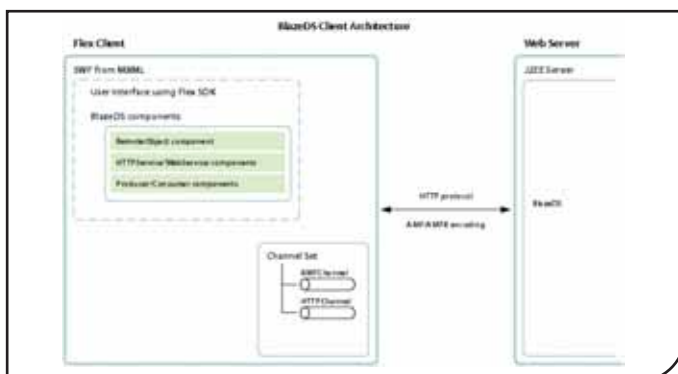
```

La destination "product" pointe sur le service Java `ProductService` en utilisant un canal de communication "my-amf" qui transporte sur le port HTTP (ou HTTPS) des données au format AMF. Le code côté Flex est toujours aussi simple :

```

<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" >
    <mx:RemoteObject id="srv" destination="product"/>

```




```
<mx:DataGrid dataProvider="{srv.getProducts.lastResult}"
width="100%" height="100%"/>
<mx:Button label="Get Data" click="srv.getProducts()"/>
</mx:Application>
```

Notez l'attribut 'destination' du tag <mx:RemoteObject> et l'appel direct de la méthode getProducts du service Java associé. Si vous souhaitez comparez AJAX à Flex, le XML à l'AMF, testez l'application Flex Census développée par James Ward sur <http://www.jamesward.com/census/>.

Analyse des échanges de données sur le réseau

Je réutilise "Live HTTP Header" pour mesurer le poids des données qui transitent sur le réseau, cette fois-ci en AMF: **4 Ko, soit 60% de gain sur la bande passante juste pour une vingtaine d'enregistrements!** Grâce à Blaze DS, je décharge mon réseau, je reçois de la donnée typée et je n'ai pas besoin du parser du XML. Adobe se charge de développer et maintenir Blaze DS pour les applications Flex qui communiquent avec un serveur Java. Pour les autres technologies, la communauté développe des mécanismes équivalents : AMFPHP, WebOrb pour les environnements .Net, RubyAMF, etc.

<mx:Consumer>: Développer un chat en 5 minutes avec Blaze DS

Blaze DS apporte aussi une technologie de messaging (HTTP messaging comme le projet Comet). Même si cette technologie n'est pas aussi évoluée que le messaging RTMP proposé par LiveCycle Data Services, elle satisfera les développeurs web qui souhaitent développer des applications collaboratives. Il faut commencer par définir un canal de communication entre Flex et BlazeDS. Editez le fichier messaging-config.xml qui se trouve dans le répertoire WEB-INF/flex/ de votre application Java et rajoutez une destination :

```
<destination id="chat"/>
```

Désormais, nous pouvons lancer le serveur Tomcat et s'abonner à ce canal en auditeur avec le tag <mx:Consumer> comme en producteur de messages avec le tag <mx:Producer>. Voici un simple client chat que vous pourrez ensuite améliorer selon vos envies :

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml"
creationComplete="consumer.subscribe()"

<mx:Script>
<![CDATA[

import mx.messaging.messages.AsyncMessage;
import mx.messaging.messages.IMessage;
```

```
private function send():void
{
    var message:IMessage = new AsyncMessage();
    message.body.chatMessage = msg.text;
    message.body.producerName = nickname.text;
    message.body.colorText = myColorPicker.selectedColor.toString(16);
    producer.send(message);
    msg.text = "";
}

private function messageHandler(message:IMessage):void
{
    log.htmlText += "<b>" + message.body.producerName + "</b><
font color='#" + message.body.colorText + "'> " + message.body.
chatMessage + "</font>\n";
}

]]>
</mx:Script>

<mx:Producer id="producer" destination="chat"/>
<mx:Consumer id="consumer" destination="chat" message="message
Handler(event.message)"/>

<mx:Panel title="Chat" width="100%" height="100%">
<mx:TextArea id="log" width="100%" height="100%"/>
<mx:ControlBar>
    <mx:Label text="Nickname:"/>
    <mx:TextInput id="nickname" width="141"/>
    <mx:Label text="Message :"/>
    <mx:ColorPicker id="myColorPicker"/>
    <mx:TextInput id="msg" width="100%" enter="send()"/>
    <mx:Button label="Send" click="send()"/>
</mx:ControlBar>
</mx:Panel>
</mx:Application>
```

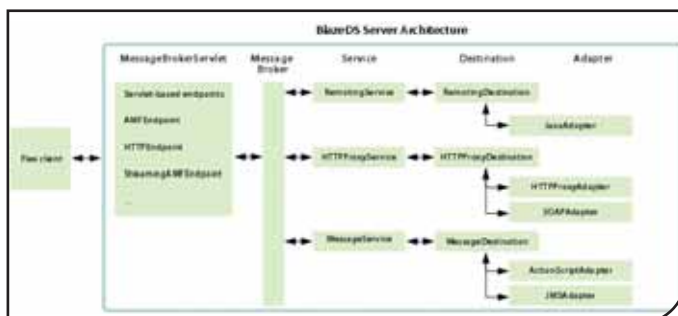
Deux tags MXML sont essentiels: <mx:Producer> et <mx:Consumer> pour émettre et recevoir des messages sur la destination "chat", le canal que nous avons créé précédemment dans le fichier XML de BlazeDS. A chaque réception d'un message, la fonction messageHandler est appelée pour générer un flux HTML constitué du nom de l'auteur et du corps du message. La fonction send() constitue un objet AsyncMessage où vous êtes libre de définir les informations à véhiculer. Ici, je transporte le nom de l'auteur, le corps du message mais aussi la couleur du texte, récupérée à l'aide du composant Color Picker. Pour tester l'application, ouvrez deux navigateurs sur la même URL et commencez à chatter.

Pour aller plus loin...

Blaze DS apporte des solutions aux développeurs Java qui souhaitent produire des RIA performantes et collaboratives. Rendez-vous sur le site <http://opensource.adobe.com> pour télécharger et tester cette solution. Blaze DS est un sous-ensemble de LiveCycle Data Services, solution payante, qui propose en plus de la génération de PDF, du messaging RTMP, de la synchronisation de données et du data-paging.

■ Michaël CHAIZE

Technical Sales - Flex & LiveCycle ES - Adobe Systems France



Programmer OpenGL avec le langage Ruby

OpenGL est une librairie 3D portable. Ruby est un langage portable. Les deux sont sympathiques. Programmer OpenGL avec Ruby est une activité sympathique également. Nous nous y adonnons ensemble.



OpenGL est une spécification qui définit une API multi-plate-forme pour la conception d'applications générant des images 3D.

Cette librairie 3D est en

concurrence avec l'API Direct 3D de Microsoft. De moins en moins utilisée dans le monde des jeux sous Windows, elle reste encore très employée dans les applications industrielles, scientifiques ou artistiques ou même dans l'embarqué. L'atout majeur d'OpenGL devant Direct3D est sa portabilité et si Direct3D règne maintenant sur les systèmes Windows, OpenGL reste l'API 3D disponible sur toutes les plates-formes, Windows inclus. Se former à OpenGL est donc une bonne idée. Comme l'immense majorité des API, OpenGL est au départ censée être programmée avec C (ou C++). Ceci pour des raisons historiques et aussi pour d'évidentes raisons liées aux performances. Cependant programmer en C devient vite pénible.

Le code peut être verbeux, manipuler les types est fastidieux et passer par le cycle écriture-compilation-édition de liens, coûteux en temps. Chaque fois que la vitesse d'exécution n'est pas critique ou lorsque le temps de développement l'est, il est intéressant de se servir d'un langage interprété plutôt que de C. Par exemple, lorsqu'on code des essais pour se familiariser avec l'API (premier cas de figure), où lorsqu'on met au point une scène complexe (second cas de figure). Ceci posé, quel langage choisir ? Un langage de haut niveau à la syntaxe claire qui permet d'écrire du code rapidement. Ainsi Python serait un excellent choix. Ruby, langage de plus en plus à la mode, est un choix excellent aussi. Nous allons utiliser celui-ci dans cet article dont le but est double: montrer à ceux qui programment OpenGL avec C comment procéder avec Ruby, et à ceux qui connaissent Ruby, montrer les rudiments de la programmation 3D avec OpenGL.

1 OUTILS ET INSTALLATION

Il est évident que le langage Ruby doit être installé sur votre système. A priori ceci est une simple formalité quelle que soit votre plate-forme de travail. Les sources de l'extension OpenGL pour Ruby peuvent être téléchargées à <http://ruby-opengl.rubyforge.org/>. Cependant, plutôt que de compiler et d'installer à partir des sources, le plus simple est certainement de passer par l'exemple utilitaire Rubygems. L'installation de ce dernier est, a priori et comme pour Ruby, une simple formalité. Après quoi, il suffit de taper dans un terminal la commande suivante :

```
gem install --remote ruby-opengl
```

Remarque : il est pertinent de faire cette opération même si vous êtes sous Windows. En effet la distribution Ruby pour cette plate-forme contient l'extension OpenGL, mais dans une version antédiluviennne. Passer par Rubygems vous assure de disposer de la toute dernière version stable, sans vous soucier le moins du monde de la présence d'une version antérieure :) Ne vous en privez surtout pas. Notre code a été écrit sous Windows Vista avec l'éditeur SciTE et

testé avec succès sous Gentoo Linux. Nous avons utilisé les dernières versions de Ruby et de ruby-opengl disponibles au moment de la rédaction de cet article, à savoir Ruby 1.8.6 et ruby-opengl 0.60.0.

2 LA LIBRAIRIE GLUT

OpenGL spécifie comment une scène 3D doit se rendre, mais elle ne spécifie pas dans quoi, fenêtre ou plein écran... C'est aux librairies qui implémentent OpenGL de traiter la question de savoir dans quoi une scène est rendue. Ainsi Qt ou wxWidgets proposent-elles un widget OpenGL qu'il est possible d'intégrer à une application. Lorsque l'on parle d'OpenGL hors d'un tel contexte il est d'usage d'employer la librairie GLUT (prononcer glout comme dans glouton :) Il s'agit d'une librairie très simple mais suffisante pour accueillir des applications "tout OpenGL". Son atout majeur est qu'elle dispense totalement de se préoccuper du système d'exploitation sous-jacent et permet de se concentrer sur la 3D. De ce fait GLUT est particulièrement appréciée lorsqu'il s'agit d'écrire des programmes de démonstration portables. Le paquet ruby-opengl intègre une implémentation de GLUT.

3 UTILISATION BASIQUE DE GLUT

Voici le code qui demande à GLUT d'ouvrir une fenêtre tellement vide qu'elle en est transparente (voir ci-dessous). [Fig.1].

```
require 'glut'
```

```
WIDTH = 300
```

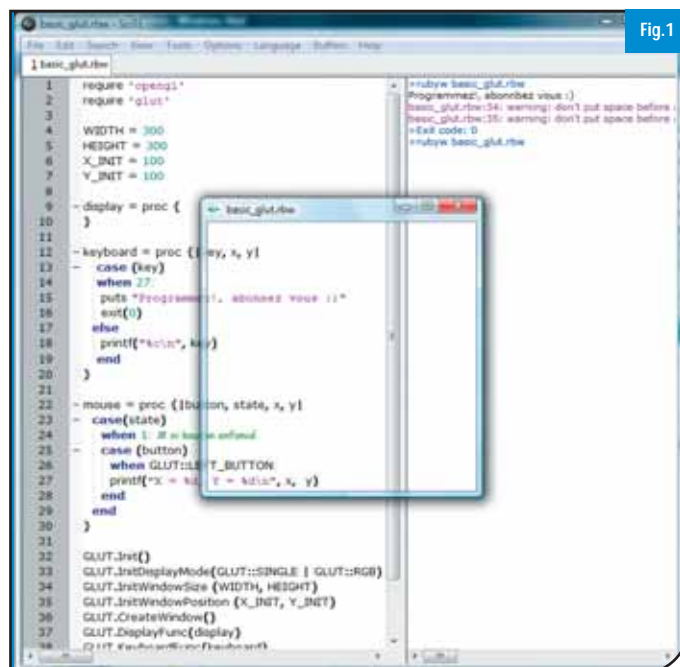


Fig.1

Ouverture d'une fenêtre Glut basique.

```

HEIGHT = 300
X_INIT = 100
Y_INIT = 100

display = proc {
}

keyboard = proc {|key, x, y|
  case (key)
  when 27:
    puts "Programmez!, abonnez vous :)"
    exit(0)
  else
    printf("%c\n", key)
  end
}

mouse = proc {|button, state, x, y|
  case(state)
  when 1: # si bouton enfoncé
    case (button)
    when GLUT::LEFT_BUTTON:
      printf("X = %d, Y = %d\n", x, y)
    end
  end
}

GLUT.Init()
GLUT.InitDisplayMode(GLUT::SINGLE | GLUT::RGB)
GLUT.InitWindowSize(WIDTH, HEIGHT)
GLUT.InitWindowPosition(X_INIT, Y_INIT)
GLUT.CreateWindow()
GLUT.DisplayFunc(display)
GLUT.KeyboardFunc(keyboard)
GLUT.MouseFunc(mouse)
GLUT.MainLoop()

```

L'exécution de ce code commence par l'appel à la fonction `GLUT.Init()` dont la présence est toujours requise. Les quatre appels de fonctions suivants s'occupent de l'ouverture de la fenêtre et ne nécessitent pas de commentaires particuliers. En revanche, les trois suivants méritent toute notre attention car ils installent les fonctions de rappel (callback) qui donneront vie à l'application. Tandis qu'en C on passe un pointeur de fonction pour cela, en Ruby on crée un objet procédure au moyen du mot-clé `proc` :

```

display = proc {
}

```

Nous aurions pu aussi utiliser la construction `Proc.new` totalement équivalente. En l'état notre procédure ne fait strictement rien, ce qui explique le caractère très dépouillé de l'affichage de notre fenêtre :) Bien sûr on se doute que tout le code de rendu d'une scène devra être placé dans cette procédure. Ensuite vient l'installation de la procédure de rappel permettant de réagir aux événements clavier. Contrairement à la fonction de rappel pour l'affichage qui ne reçoit jamais rien, les paramètres reçus par la fonction de rappel du clavier sont au nombre de trois. Le premier est le code ASCII de la touche actionnée. Les deux autres correspondent à la position, en coordonnées fenêtre, du pointeur de souris au moment de l'action sur la

touche. Notre exemple montre comment arrêter l'application lorsque l'utilisateur enfonce la touche `Escape` (code ASCII 27), ce qui est la manière traditionnelle de procéder sous GLUT. En effet, il arrive que des implémentations GLUT ne se comportent pas parfaitement lorsque l'utilisateur clique sur le bouton de fermeture de la fenêtre. Ensuite, notre programme d'exemple installe la procédure de rappel qui permet de réagir aux événements souris. Le code parle de lui-même et nous ne le commentons pas. Enfin, nous lançons la boucle d'événements par l'appel `GLUT.MainLoop()`. Cet appel doit être présent dans toute application GLUT propre sur elle. Pour en finir avec GLUT, signalons que nous n'avons pas fait le tour de toutes ses possibilités. Même si GLUT est simple en comparaison d'autres toolkit, elle est malgré tout riche. Le lecteur qui voudrait en savoir plus, par exemple comment réagir aux actions sur les touches spéciales du clavier, pourra consulter la documentation de l'API GLUT à <http://www.opengl.org/documentation/specs/glut/spec3/spec3.html>

4 SYSTÈME DE COORDONNÉES ET VIEWPORT OPENGL

Il est temps maintenant de commencer à rendre une scène dans notre fenêtre `Glut`. Notre premier objectif est très modeste: dessiner un carré dans le plan de la fenêtre. [Fig.2] Si le résultat à un look 2D, nous travaillons pourtant en 3D, mais dans un cas simple car il s'agit tout d'abord de bien comprendre comme est cadrée une scène. OpenGL utilise un système de coordonnées orthonormé direct. L'axe des `x` est orienté de gauche à droite, l'axe des `y` de bas en haut et l'axe des `z` sort de l'écran pour pointer vers vous. Ceci posé, où se situe l'origine du repère ? Là où le veut le programmeur. Pour cela on appelle la fonction OpenGL `glViewport`, ce qui en Ruby se traduit par `GL.Viewport`. Le principe de traduction est simple, les préfixes des fonctions C/OpenGL trouvent leur équivalent dans un nom de module Ruby. `GL.Viewport` définit à la fois la position de l'origine et la taille utile de la fenêtre. Dans le code donné plus bas, nous trouvons :

```
GL.Viewport(0, 0, w, h)
```

Les paramètres reçus par la fonction sont des coordonnées fenêtre, puisque le repère orthonormé n'est pas encore positionné :) ici l'origine du repère est placée en bas et à gauche de la fenêtre, et nous utilisons la fenêtre entière. Nous pouvons expérimenter. Par exemple :

```
GL.Viewport(w/2, h/2, w, h)
```

Cet appel place l'origine du repère au centre de la fenêtre. Le résultat [Fig.3] montre bien que le rendu de notre carré est traduit en conséquence. Une dernière remarque, dans le code ci-dessous, l'appel à `GL.Viewport` est effectué dans une fonction de rappel baptisée `reshape`. C'est le bon endroit, car cette fonction de rappel est



Fig.2
Dessin d'un carré, dans le plan d'une fenêtre `Glut`.

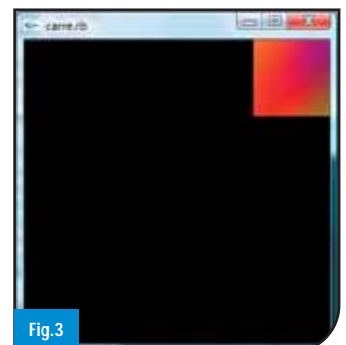


Fig.3
Changer l'origine du repère translate le rendu d'une scène dans la fenêtre `Glut`.

invoquée si l'utilisateur redimensionne la fenêtre, ce qui permet de redéfinir le viewport à la volée.

5 MATRICES DE PROJECTION

Sans entrer dans les détails mathématiques ardues, disons qu'une matrice de projection a pour rôle de "plaquer" une scène 3D sur notre fenêtre, pour finalement obtenir un dessin en 2D, la seule chose que l'ordinateur peut représenter, puisque l'écran ne possède que deux dimensions :) Il existe deux matrices de projection avec OpenGL: la matrice de projection conique que l'on définit par un appel à `GL.Frustum` et la matrice de projection orthogonale que l'on définit par un appel à `GL.Ortho`. Cette dernière a pour propriété que les faces parallèles au plan xy demeurent inchangées (projection en perspective cavalière). C'est donc elle que nous utilisons pour rendre notre carré. Dans la fonction de rappel `reshape`, nous invoquons la fonction ainsi:

```
GL.Ortho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0)
```

Cet appel signifie que l'axe des x sur la largeur de la fenêtre est le vecteur unitaire. Même chose pour l'axe des y. Nous ne parlons pas des deux derniers paramètres (-1.0, et 1.0) qui concernent la profondeur. `GL.Ortho` est pleine de surprises. Il est important de bien comprendre le principe en s'entraînant un petit peu, sous peine de voir disparaître notre scène "quelque part". Ainsi demandons nous quel serait le résultat si on remplaçait par:

```
GL.Ortho(-1.0, 1.0, -1.0, 1.0, -1.0, 1.0)
```

L'axe des x s'étend maintenant de -1.0 à 1.0, donc nous avons déplacé implicitement l'origine du repère pourtant positionnée en amont par `GL.Viewport` :) Notre carré est-il donc finalement translaté comme dans l'exemple précédent. Pas tout à fait.... En effet les axes ayant une longueur double, et les dimensions de l'objet étant inchangées, son rendu est réduit (et non grossi :) de moitié, et on obtient le résultat illustré ci-contre. [Fig.4].

6 DÉFINITION DU CONTENU D'UNE SCÈNE

Mais comment diable notre carré est-il défini finalement ? Il l'est dans la fonction de rappel `display` et dans un sandwich OpenGL dont les deux tranches de pain sont les incontournables appels `GL.Begin` et `GL.end`. Entre ces deux là, le code ci-dessous définit des couleurs pour chaque point, ou comme on dit en trois 3D chaque vertex, c'est-à-dire chaque sommet d'un polygone. Les coordonnées sont définies de manière évidente par rapport à notre viewport. C'est ainsi qu'il convient de procéder pour éviter de se plonger dans d'insurmontables difficultés. Si l'objet doit être réduit ou déplacé dans le viewport, ce sont les matrices de projections qui vont se charger de la besogne, et surtout pas le programmeur. Dernière remarque, puisque dans l'exemple nous travaillons en double-buffering, c'est l'appel à `GLUT.SwapBuffers` qui provoque l'affichage. Voici maintenant le code partiel de notre exemple:

```
require 'opengl'
require 'glut'

WIDTH = 300
HEIGHT = 300
X_INIT = 100
```

```
Y_INIT = 100
```

```
def init
  GL.ClearColor(0.0, 0.0, 0.0, 0.0)
  GL.ShadeModel(GL::SMOOTH)
  GL.MatrixMode(GL::MODELVIEW)
  GL.LoadIdentity()
  GL.Ortho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0)
  #GL.Ortho(-1.0, 1.0, -1.0, 1.0, -1.0, 1.0)
end
```

```
reshape = proc { |w, h|
  GL.Viewport(0, 0, w, h)
  #GL.Viewport(w/2, h/2, w, h)
}
```

```
display = proc {
  GL.Clear(GL::COLOR_BUFFER_BIT)
  GL.Begin(GL::POLYGON)
  GL.Color3f(1.0, 0.0, 0.0)
  GL.Vertex3f(0.25, 0.25, -0.0)
  GL.Color3f(0.0, 1.0, 0.0)
  GL.Vertex3f(0.75, 0.25, 0.0)
  GL.Color3f(0.0, 0.0, 1.0)
  GL.Vertex3f(0.75, 0.75, 0.0)
  GL.Color3f(1.0, 1.0, 0.0)
  GL.Vertex3f(0.25, 0.75, 0.0)
  GL.End()
  GLUT.SwapBuffers()
}
```

```
GLUT.Init()
GLUT.InitDisplayMode(GLUT::DOUBLE | GLUT::RGB | GLUT::DEPTH)
GLUT.InitWindowSize(WIDTH, HEIGHT)
GLUT.InitWindowPosition(X_INIT, Y_INIT)
GLUT.CreateWindow()
init()
GLUT.DisplayFunc(display)
GLUT.ReshapeFunc(reshape)
GLUT.KeyboardFunc(keyboard)
GLUT.MouseFunc(mouse)
GLUT.MainLoop()
```

7 UNE SCÈNE 3D ANIMÉE

Que nous manque-t-il pour avoir une scène avec un vrai look 3D, et animée qui plus est ? Très peu de choses. `GLUT.SwapBuffers` est synchronisée avec le balayage écran et nous travaillons en double tampon, ce qui nous garantit la fluidité de l'animation. Pour obtenir réellement une animation, il faut invoquer la procédure d'affichage en boucle infinie. Pour cela nous installons une procédure idle qui poste un message demandant l'affichage chaque fois que la boucle des messages est vide. A chaque passage, il suffit d'incrémenter, par exemple une valeur d'angle et une valeur d'éloignement, pour respectivement faire tourner un objet (ici un simple triangle) autour de son axe vertical et le déplacer d'avant en arrière. [Fig.5] Les rotations et translations sont obtenues tout simplement avec les fonctions `GL.Translate` et `GL.Rotate` appliquées à l'ensemble des points définissant le triangle et pour obtenir cette fois un effet réaliste,

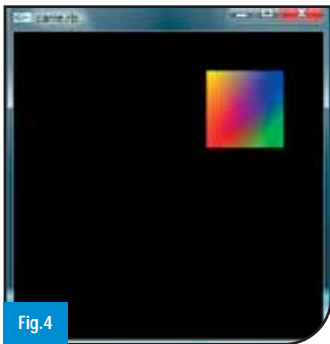


Fig.4 Une matrice de projection influe à la fois sur la position et l'échelle d'un objet.

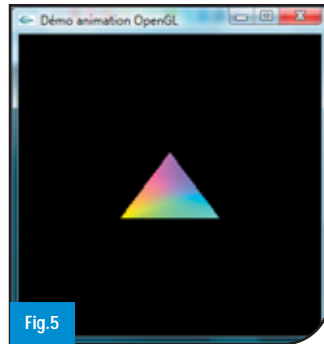


Fig.5 Un triangle animé de deux mouvements simultanés.

nous utilisons une matrice de projection conique (fonction GL.Frustum) plutôt qu'une matrice de projection orthogonale. Voici le code partiel :

```
require 'opengl'
require 'glut'
# si utilisation des APIs GLU
require 'glu'
WIDTH = 300
HEIGHT = 300
X_INIT = 100
Y_INIT = 100

$liste_affichage = 0
$angle = 0.0
$eloignement_maxi = -49.0
$eloignement_mini = -12.0
$eloignement = $eloignement_mini
# Un pas trop grand provoque des saccades
# quand l'objet est trop près de l'objectif
$pas = 0.2
$sens = true
draw = proc {
  GL.Clear(GL::COLOR_BUFFER_BIT)

  GL.LoadIdentity()
  GL.Translate(0.0, 0.0, $eloignement) # !!! et non GL.Rotatef... !!!
  # ou:
  # GLU.LookAt(0.0, 0.0, -eloignement, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0)

  # Rotation pour le triangle
  GL.Rotate($angle, 0.0, 1.0, 0.0)
  GL.Begin(GL::TRIANGLES)
    GL.Color3f(1.0, 0.0, 1.0)
    GL.Vertex3f(0.0, 0.5, 0.0)
    GL.Color3f(1.0, 1.0, 0.0);
    GL.Vertex3f(-0.75, -0.5, 0.0)
    GL.Color3f(0.0, 1.0, 1.0)
    GL.Vertex3f(0.75, -0.5, 0.0)
  GL.End()
  # fin transformation de modélisation

  GLUT.SwapBuffers()

  $angle = $angle + 1.0
```

```
if $angle > 360.0
  $angle = 0.0
end

if $sens == true
  $eloignement -= $pas
  if $eloignement < $eloignement_maxi
    $sens = false
  end
else
  $eloignement += $pas
  if $eloignement > $eloignement_mini
    $sens = true
  end
end

# fin proc draw :)
}

idle = proc {
  GLUT.PostRedisplay()
}

reshape = proc { |width, height|
  GL.Viewport(0, 0, width, height)
  # Transformation de projection
  GL.MatrixMode(GL::PROJECTION)
  GL.LoadIdentity()
  GL.Frustum(-1.0, 1.0, -1.0, 1.0, 10.0, 50.0)
  # Transformation de visualisation
  GL.MatrixMode(GL::MODELVIEW)
}

def init
  GL.ClearColor(0.0, 0.0, 0.0, 0.0)
  GL.ShadeModel(GL::SMOOTH)
end

GLUT.Init()
GLUT.InitDisplayMode(GLUT::RGB | GLUT::DEPTH | GLUT::DOUBLE)
GLUT.InitWindowPosition(X_INIT, Y_INIT);
GLUT.InitWindowSize(WIDTH, HEIGHT);
GLUT.CreateWindow("Démon animation OpenGL")
init()

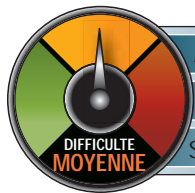
# Pour un affichage plein écran
# GLUT.FullScreen()
GLUT.DisplayFunc(draw)
GLUT.ReshapeFunc(reshape)
GLUT.KeyboardFunc(key)
GLUT.SpecialFunc(specialkey)
GLUT.VisibilityFunc(visible)
GLUT.MainLoop()
```

La programmation OpenGL avec Ruby est loin de nous avoir livré tous ses secrets, tant en ce qui concerne OpenGL elle-même qu'en ce qui concerne Ruby dans ce contexte d'utilisation. Nous nous retrouverons donc bientôt.

■ Frédéric Mazué - fmazue@programmez.com

Java 6 Update 10 : en attendant Java 7

Contrairement aux changements majeurs de version, les mises à jour du langage Java doivent respecter un certain nombre de contraintes visant à laisser intactes les API publiques du langage. Ainsi, une mise à jour ne peut venir ajouter, modifier ou supprimer le contenu de ces dernières. De ce fait, les updates du langage se limitent bien souvent à la correction de bugs.



APPLICATION : DESKTOP

LANGAGE : JAVA

SOURCE : NON

A contrario, cette update va bien plus loin et se révèle comme une des plus intéressantes proposée par Sun en apportant de nouvelles fonctionnalités orientées utilisateur final qui vont faciliter le déploiement et l'exécution d'applets.

JAVA KERNEL

La JRE de Java 6 met à disposition un grand nombre d'API de tous types parmi lesquelles on peut citer de manière non exhaustive Swing, AWT, SQL, CORBA, RMI, math, XML, concurrency, ... Le nombre important de ces API implique qu'elles ne sont pas forcément toutes utilisées en même temps par votre programme Java. De ce postulat est né le projet Java Kernel qui est une version allégée de la JRE réduite aux composants de base de Java. La taille de ce noyau de base est très proche de celle de la première JRE comme le montre la figure 1. Dans un premier temps, seule l'installation de ce noyau sur le poste client est nécessaire.

La force du Java Kernel réside ensuite dans le fait que lors de l'exécution d'une application, les composants additionnels nécessaires sont téléchargés de manière transparente pour l'utilisateur final. Le Kernel de base présent sur le système de l'utilisateur se retrouvant ainsi complété au fur et à mesure de l'exécution d'applications Java. Le site de Sun indique que pour une application de type applet utilisant Swing, le Kernel complété des différents composants complémentaires nécessaires atteindra une taille comprise entre 4 et 5 Mo, ce qui reste encore bien inférieur à la taille de la JRE 6. Ceci permettant de réduire le temps de démarrage des applications Java. Ainsi, le déploiement d'applets sur le poste client va gagner en rapidité, de même que le déploiement d'applications Java via la technologie Java Web Start. Ce gain de performance était nécessaire face à l'émergence de technologies telles que Silverlight ou Flex.

NEXT GENERATION JAVA PLUG-IN

Le déploiement des applets restait jusqu'à présent quelque chose d'assez compliqué avec des problématiques de gestion de version de la JRE et d'exécution de cette dernière au sein du client léger servant de support au déploiement.

Le module Java Plug-In apporté par Java 6 Update 10 vient répondre à cette problématique. Inclus dans la JRE, il est disponible pour Internet Explorer mais également pour Firefox 3. En premier lieu, il permet d'exécuter l'applet au sein d'un ou plusieurs processus séparés du navigateur dans lequel elle est visible. L'applet pouvant ensuite être détachée du navigateur pour être utilisée comme une application Java à part entière et indépendante. Ceci permettant d'accroître la stabilité des applets.

En outre, afin de permettre une certaine uniformisation dans le mode de lancement d'applications Java depuis un client léger, le module Java Plug-In offre la possibilité de configurer le déploiement d'applets, via la technologie Java Web Start (JWS) à l'aide de fichiers JNLP (Java Network Launching Protocol). Cet effort d'unification a pour première conséquence de rendre accessible aux applets un certain nombre de services présents dans l'API JNLP et plus particulièrement :

- DownloadService pour la mise en cache des ressources utilisées par une application
- FileOpenService et FileSaveService qui fournissent des méthodes pour ouvrir et écrire des fichiers depuis le disque local, et ce, même pour des applications s'exécutant dans un environnement d'exécution restreint au niveau sécurité
- PersistenceService qui fournit des méthodes pour le stockage d'informations sur le poste client

Le code suivant détaille le lancement de l'applet Sample via un fichier JNLP :

```
<applet width="150" height="60" codebase="file:///I:/divers/articles/j6u10/">
  <param name="jnlp_href" value="Sample.jnlp"/>
  <param name="draggable" value="true" />
  <param name="name" value="Programmez !"/>
</applet>
```

L'attribut codebase du tag *applet* définit le chemin vers le fichier JNLP qui lui est défini au sein d'un tag *param* ayant une propriété valant *jnlp_href*. Le positionnement de la propriété *draggable* à *true* permet d'activer le drag and drop de l'applet depuis le navigateur. Le fichier JNLP *testApplet.jnlp* ayant une forme assez classique, si ce n'est l'ajout du tag *applet-desc* pour spécifier l'applet à lancer ainsi que ses paramètres de lancement. Le voici :

```
<?xml version="1.0" encoding="UTF-8"?>
<jnlp codebase="file:///I:/divers/articles/j6u10/" href="Sample.jnlp">
  <information>
```

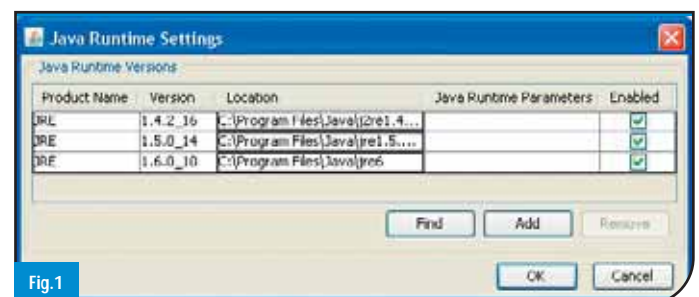


Fig.1

DÉVELOPPEZ 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 !

ATTENTION !
Tarifs en hausse le mois prochain !

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
- **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**

*Tarif France métropolitaine

+ Abonnement INTÉGRAL

ACCÈS ILLIMITÉ aux ARCHIVES du MAGAZINE pour 1€ par mois !

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

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

PROgrammez !
Le magazine du développement

Offre limitée,
valable jusqu'au
31 décembre 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.

```

<title>Sample</title>
<vendor>Programmez ;</vendor>
<offline-allowed />
</information>
<!-- Permet d'activer la création d'un raccourci sur le bureau
pour l'applet -->
<shortcut online="false">
  <desktop />
</shortcut>
<resources>
<j2se version="1.6+" href="http://java.sun.com/products/aut
odl/j2se" />
<!-- Ajout d'extensions éventuelles (vers des Jar ou d'autres
JNLP) -->
<jar href="Sample.jar" />
</resources>
<!-- Les attributs width / height définis ici sont nécessaires
pour le lancement de l'applet depuis le bureau -->
<applet-desc name="Sample"
  main-class="Sample"
  width="150"
  height="60">
</applet-desc>
</jnlp>

```

Cette applet, somme toute banale, qui se contente d'afficher la date courante (figure 2) profite pleinement des nouveautés du plug-in Java apportées par Java 6 Update 10 puisqu'il est possible de la détacher du navigateur dans lequel elle a été lancée en utilisant la série de commandes suivantes : Alt + Clic gauche + Drag.

De plus, lorsque l'on dépose l'applet hors du navigateur via un drag and drop, cette dernière se retrouve déconnectée du navigateur, ce qui implique qu'elle ne peut plus communiquer avec lui et c'est à ce moment là que la JRE vous propose de créer un raccourci sur le bureau qui vous permettra de relancer cette applet comme un programme Java indépendant.

Les API publiques Java ne pouvant être modifiées lors d'une update, les fonctions permettant d'interagir avec l'applet durant le drag and drop n'ont pas pu être intégrées dans une classe spécifique mais elles sont d'ores et déjà utilisables à titre expérimental. L'applet Sample, dont le code est disponible sur le CD ou notre site, met en œuvre toutes ces possibilités.

JAVA DEPLOYMENT TOOLKIT

Le mode de déploiement d'applets ou d'applications via JWS présenté précédemment nécessite un niveau de configuration assez important tant au niveau de la page HTML qu'au niveau d'éventuels fichiers de lancement JNLP. Afin d'améliorer ces points négatifs,

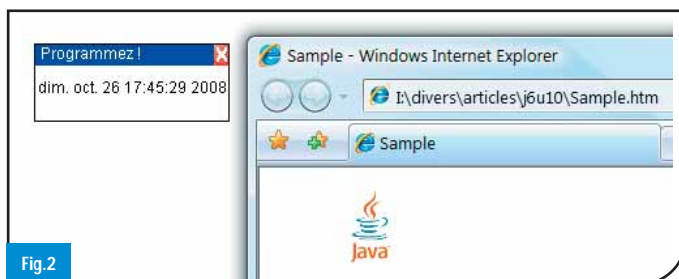


Fig.2

Applet Sample déposée à l'extérieur du navigateur

cette update 10 met à disposition du développeur une classe utilitaire Javascript nommée *deployJava* qui au travers d'une dizaine de fonctions publiques va :

- faciliter la configuration du déploiement des applications en réduisant les lignes de code à écrire à la portion congrue
- permettre une meilleure détection des JRE installées sur le poste client afin d'éviter d'effectuer un appel inutile à Java Web Start
- proposer le lancement de l'installation de la JRE appropriée pour l'application.
- utiliser les fonctionnalités du nouveau module Java Kernel en proposant l'ajout de nouveaux composants à l'utilisateur, le cas échéant

En reprenant l'exemple de l'applet Sample présentée un peu plus haut, on obtient le code suivant qui doit être intégré dans une page HTML :

```

<script src="http://java.com/js/deployJava.js"></script>
<script>
  var attributes = {codebase:'file:///I:/divers/articles/j6u10/',
                    code:'fr.test.Sample',
                    archive:'Sample.jar',
                    width:150, height:60} ;
  var parameters = {draggable:'true', title:'Programmez !'} ;
  var version = '1.6' ;
  deployJava.runApplet(attributes, parameters, version);
</script>

```

La méthode *runApplet* de l'objet *deployJava* prend à sa charge la détection de la JRE appropriée et propose éventuellement à l'utilisateur de se mettre à niveau le cas échéant. Elle s'occupe également de créer le tag applet approprié. Quant au lancement via un fichier JNLP, il est très aisé à réaliser et se fait comme suit :

```

<script src="http://java.com/js/deployJava.js"></script>
<script>
  var url = 'file:///I:/divers/articles/j6u10/SampleLight.jnlp';
  deployJava.createWebStartLaunchButton(url, '1.6');
</script>

```

La grosse différence ici résidant dans le contenu du fichier JNLP qui est réduit de manière drastique pour n'afficher que les informations concernant le nom du jar et la classe principale de l'applet. Les infor-



Fig.3

Composants Swing avec Nimbus

mations apportées par l'attribut codebase du tag *jnlp* sont créées ici par la méthode *createWebStartLaunchButton* de l'objet *deployJava* et sont désormais transparentes pour le développeur.

AMÉLIORATION DES PERFORMANCES

Les évolutions de la JRE ont toujours été dirigées dans le sens de l'amélioration des performances du langage Java. Cette update 10 ne déroge pas à la règle. Outre le Java Kernel présenté au début de cet article, le Java Quick Starter (JQS) fait son apparition dans cette mise à jour. JQS a pour objectif de réduire le temps de démarrage des applications Java et plus particulièrement des applets en préchargeant les fichiers les plus utilisés par la JRE dans le cache disque du système d'exploitation. Actuellement, ce nouveau module n'est disponible que pour les systèmes Windows 32 bits et est activé par défaut sur les versions 2000 et XP.

Afin de rendre homogène le niveau de performance sur l'ensemble des plates-formes d'exécution de Java, Sun a réécrit le pipeline de rendu graphique 3D pour Windows qui utilise désormais l'API Direct3D 9.0. Les applications Swing et Java 2D s'exécutant sous Windows peuvent ainsi tirer profit des possibilités d'accélération matérielles offertes par la carte graphique installée, comme cela était déjà le cas sous Linux et Mac OS X.

Dernière amélioration notable en terme de rapidité d'exécution, la compression Pack200 est désormais utilisable pour le déploiement des applets et des applications JWS sans configuration spécifique côté serveur. Conçue pour être très efficace sur les fichiers class de Java, Pack200 permet de réduire considérablement la taille des archives jar, ce qui constituera un gain de temps non négligeable lors du déploiement.

Au sein d'un fichier JNLP, il suffit de positionner la propriété *jnlp.packEnabled* à true au sein du tag ressources. Pour une applet, il faut positionner la même propriété à true dans les arguments envoyés à la JVM comme suit : `-Djnlp.packEnabled=true`.

NIMBUS LOOK AND FEEL

Les développeurs d'interface Swing vont pouvoir se réjouir avec l'arrivée tant attendue d'un nouveau Look and Feel multi-plate-forme du nom de Nimbus (figure 3). Il vient remplacer le vieillissant mais toujours vaillant Metal présent depuis la création du langage Java mais qui est aujourd'hui dépassé.

L'implémentation de Nimbus est basée sur l'outil *Synth* qui permet de créer des thèmes de manière déclarative via des fichiers XML sans manipulation de code Java. Entièrement dessiné à l'aide de graphiques vectoriels Java 2D, il a une taille réduite et peut être affiché à n'importe quelle résolution sans perte de rendu.

CONCLUSION

Outre les corrections habituelles de bugs, les améliorations apportées par Java 6 Update 10 se situent principalement au niveau de l'environnement d'exécution de Java et étaient attendues depuis un certain temps. Elles offrent ainsi une bouffée d'air frais aux applets. Enfin, les avancées en matière de performances et de déploiement étaient nécessaires à l'émergence de la future technologie orientée RIA de Sun, Java FX, qui va entrer en compétition avec les technologies Silverlight et Flex en 2009 et qui aura fort à faire.

■ Sylvain SAUREL

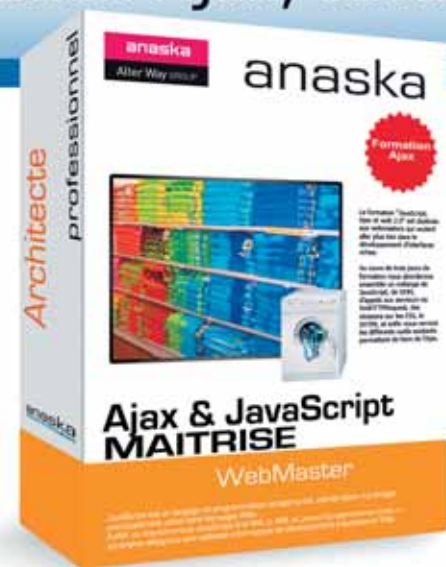
Ingénieur d'Etudes et Développement JEE

PRENEZ DE LA HAUTEUR

Web > Ajax



Formation Ajax, JavaScript



"Pour nettoyer plus blanc que blanc ! *"
** Et aussi être prêt pour le Web riche"

Au Programme des 3 jours :

- * Qu'est-ce qu'AJAX ?
- * Introduction au XML
- * Introduction au Web 2.0
- * Les feuilles de style en cascade (CSS)
- * JavaScript
- * DOM
- * Contraintes d'AjAx
- * Bibliothèques et API

10 % de réduction
pour
les lecteurs de
programmez

Prochaines sessions

Paris 2008 - 2009
22 Décembre
26 Janvier, 23 Février,
23 Mars, 20 Avril

Lyon 2008 - 2009
22 Décembre
23 Février, 23 Mars,
20 Avril, 22 Juin

Tarif
1200 € HT

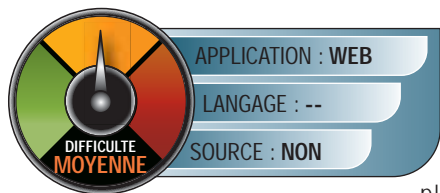
LE SPECIALISTE DE LA FORMATION POUR L'OPEN SOURCE

anaska
formation aux logiciels libres
www.anaska.com
Alter Way GROUP

Informations
01 45 28 09 82
www.anaska.com

Déconnectez votre application avec Google Gears

Gears est sorti depuis maintenant plus d'un an. Peu d'applications en font aujourd'hui usage et pourtant cela permet d'améliorer considérablement les performances et le confort utilisateur. Voyons de près comment faire et réalisons une application de gestion de vos contacts Google pouvant travailler en mode déconnecté.



SYNCHRONISER

Cette étape est de loin la plus compliquée. La synchronisation est ce qui donne le plus de fil à retordre pour nombre d'applications aujourd'hui. Pensez à la synchronisation des contacts de votre iPhone avec les contacts de votre application Mac Contacts et vos contacts Google et vos contacts Outlook. Argh ! Par ordre de complexité les scénarios peuvent être les suivants :

- cache en lecture seule
- cache en lecture seule avec mise à jour incrémentale. Périodiquement, le cache local est mis à jour avec uniquement les nouvelles données où les données modifiées.
- cache en écriture pour un seul utilisateur. Le cache peut être mis à jour mais comme il est utilisé par un seul utilisateur les risques de conflits avec le serveur sont rares.
- cache en écriture pour de multiples utilisateurs sans fusion. Ce comportement est similaire à un CVS : si A et B modifient la même donnée, le premier qui synchronise a raison, le second devra faire une fusion en local des nouvelles données avec les siennes avant de resoumettre.
- cache en écriture pour de multiples utilisateurs avec fusion. Il s'agit du même cas que précédemment mais cette fois le serveur tente de faire une fusion automatique des données avec gestion des conflits.

Qu'avons-nous fait jusqu'à présent ? Si on repense à la notion de stratégie définie plus avant, nous avons donc réalisé jusqu'à présent une application modale avec cache local en lecture seule. C'est le cas le plus simple. Mais ceci permet déjà d'accélérer l'application et de faire des recherches plein texte ! Passons tout de même à la

vitesse supérieure et mettons à jour les données en mode déconnecté. Pour cela il nous faut d'abord suivre l'état dans lequel l'application se trouve en ajoutant une variable globale 'offline' qui nous dira en permanence si l'application est déconnectée ou non.

En mode connecté, toutes les modifications faites sur l'interface seront immédiatement envoyées au serveur et dans le même temps feront l'objet d'une mise à jour en base. En mode déconnecté, les modifications seront placées dans la base uniquement et un flag 'updated' sera posé sur l'enregistrement.

```
afteredit: {
  fn: function(o){
    var db = google.gears.factory.create('beta.database');
    db.open('gcontacts');
    if (!offline) {
      var entry = o.record.data.rawObj;
      if (o.field == 'title') {
        updateTitle(entry, o.value);
        db.execute('update contacts set name="' + o.value + '"
          where email="' + o.record.get('primaryEmail') + '"');
```

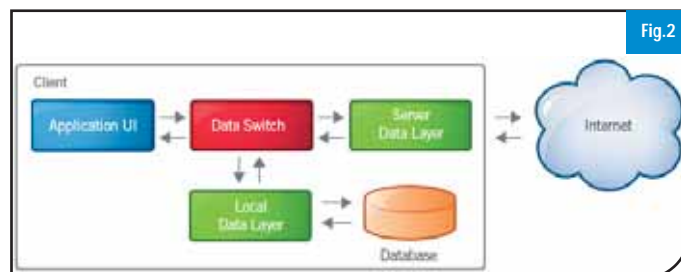


Fig.2

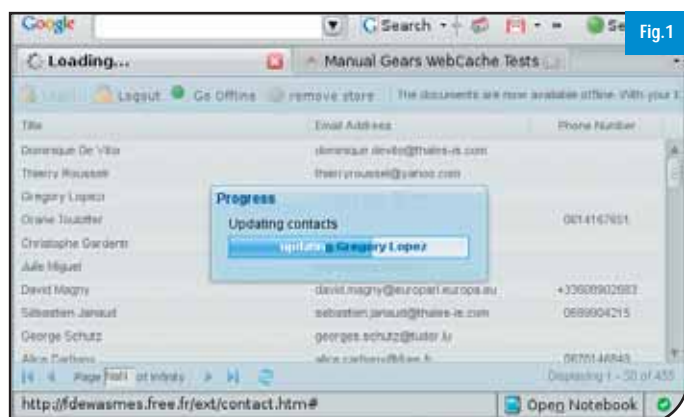


Fig.1



Fig.3

```

    }
[...]
```

```

    // Commit Changes
    entry.updateEntry(function(){
        Ext.getCmp('gc-grid').store.commitChanges();
    }, GContact.errorHandler);
    } else {
        if (o.field == 'title') {
            db.execute('update contacts set updated=1, titleupdate=1, name="' + o.value + '" where email="' + o.record.get('primaryEmail') + '"');
        }
    }
[...]
```

```

    Ext.getCmp('gc-grid').store.commitChanges();
    }
}
}
```

Le passage du mode connecté au mode déconnecté est transparent mais il faut encore traiter le passage inverse en envoyant au serveur les données modifiées localement. [Fig.1]

Pour cela, nous allons créer une fonction resync qui va chercher en base les données modifiées et envoyer des requêtes de modification au serveur :

```

function resync(uri, contactservice){
    var db = google.gears.factory.create('beta.database');
    db.open('gcontacts');

    rs = db.execute('select * from contacts where updated=1');
    while (rs.isValidRow()) {
        var cs = contactService;
        cs.getContactEntry(rs.field(4), handleContactUpdate(result));
        rs.next();
    }
    rs.close();

    var onlinestore = Ext.StoreMgr.lookup('online');
    Ext.getCmp('pager').bind(onlinestore);
    Ext.getCmp('gc-grid').reconfigure(onlinestore, columnModel);
    onlinestore.load();
    offline = false;
}
}
```

Ici, pour chaque enregistrement en base, le contact correspondant est récupéré du serveur et passé à la méthode handleContactUpdate qui va copier les modifications de la base dans le contact, envoyer les modifications au serveur et mettre à jour la base en faisant un reset du flag 'updated'. Nous avons ainsi implémenté ce joli schéma de Google [Fig.2].

LA CERISE SUR LE GÂTEAU

Ajoutons à présent deux petites choses presque pour le plaisir. Nous allons maintenant permettre à l'utilisateur d'installer une icône

de lancement de notre belle application dans son système. Pour cela nous utilisons les fonctionnalités Desktop de Gears en ajoutant ce code lors de la création du serveur de fichiers local :

```

var desktop = google.gears.factory.create('beta.desktop');

desktop.createShortcut('Google Contacts Application',
    'http://<serv_root>/contact.htm', {
    '128x128': 'http://<serv_root>/icons/addressbook128.png',
    '48x48': 'http://<serv_root>/icons/addressbook48.png',
    '32x32': 'http://<serv_root>/icons/addressbook32.png',
    '16x16': 'http://<serv_root>/icons/addressbook16.png'},
    'A wonderful to manage your contacts');
```

Ceci demandera à l'utilisateur s'il accepte d'installer un raccourci. Le premier paramètre est le nom de l'application, le second, l'URL de l'application, le troisième, un objet donnant pour chaque taille l'URL d'une icône et le dernier est une description de l'application. [Fig.3]

Enfin, ajoutons une fonction qui va périodiquement vérifier l'état de connexion du navigateur pour nous éviter d'avoir à faire une bascule manuelle entre les deux modes :

```
var TIME_BETWEEN_PINGS = 3 * 1000;
```

```

function isServerAvailable(){
    if (!navigator.onLine) if (!offline) goOffline(false);
    else if (offline) resync();
    window.setTimeout("isServerAvailable()", TIME_BETWEEN_PINGS);
}
```

Il ne reste plus qu'à lancer la fonction au démarrage de l'application. Dès que vous passerez votre navigateur en mode hors ligne, l'application basculera automatiquement en mode déconnecté. Bien sûr cette approche est simpliste, et il faudrait également tester si des requêtes HTTP retournent en erreur.

CONCLUSION

Voilà ! Je vous invite à refaire l'application en partant de celle d'ExtJS. Nous avons vu dans cet article quasiment toutes les fonctionnalités de Gears et surtout comment rendre votre application déconnectée. J'espère que cela vous aura permis de faire un premier pas pour passer votre propre application en mode déconnecté. L'application est visible online à l'adresse : <http://fdewasmes.free.fr/demo/gears/>. Les sources de l'application sont disponibles sur mon blog.

Ressources :

<http://extjs.com/playpen/GContact/GContact.html>
http://code.google.com/apis/gears/articles/take_app_offline.html
<http://code.google.com/apis/gears/>
<http://extjs.com/>
<http://sqlitestudio.one.pl/>

■ Fabrice Dewasmes

Blog : <http://fdewasmes.free.fr>

Fabrice.dewasmes@gmail.com

ABONNEMENT PDF : 2,73 € le numéro

www.programmez.com

Python



Difficulté : **
 Editeur : Eni éditions
 Auteur : Matthieu Brucher
 Prix : 27,14 €

Vous n'êtes pas encore un adepte de Python ? Voici une bonne occasion de vous y mettre avec ce livre qui présente les fondamentaux du langage et la programmation orientée scientifique. L'auteur décrit rapidement le langage puis s'attaque aux modules de la bibliothèque standard et des outils purement scientifiques et mathématiques. Point intéressant : une partie entière consacrée à l'interfaçage C/C++ avec Python.

Les design patterns en Ruby



Difficulté : ****
 Editeur : Pearson
 Auteur : collectif
 Prix : 35 €

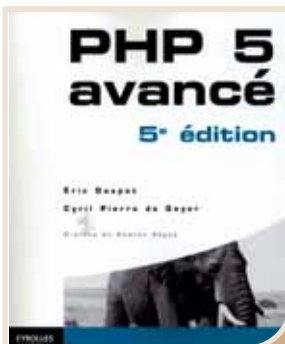
On l'oublie souvent, mais en Ruby nous pouvons mettre en œuvre des patrons ou design pattern comme on peut le faire en .Net, java, C++. L'ouvrage met le pied dans le plat avec la présentation, à fond, de 14 des 23 patterns les plus utilisés, certains ayant été spécialement conçus par la communauté Ruby. Malgré le sujet, parfois difficile, le livre est toujours clair et compréhensible. Tout développeur Ruby devrait posséder cet ouvrage de référence.

Silverlight 2



Difficulté : **
 Editeur : Eyrolles
 Auteur : Gérard Leblanc
 Prix : 27,14 €

Silverlight 2 est la vraie version que l'on aurait dû posséder dès le départ : langage dynamique, intégration de .Net, meilleure intégration avec les outils, nouveaux objets, etc. Silverlight 2 a tout pour plaire et surtout pour contrer le couple Flex / Air d'Adobe. L'auteur nous prend par la main et démontre le processus de développement : comment créer une application, la gestion



PHP 5 avancé 5^e édition

Difficulté : ***
 Editeur : Eyrolles
 Auteur : Eric Daspet, Cyril Pierre de Geyer
 Prix : 45 €

On aurait peut être dû le nommer *PHP, la totale* car il ne faut pas se fier à son titre : le livre PHP 5 Avancé convient aussi bien aux débutants qu'aux utilisateurs chevronnés ! Il s'attaque tout d'abord aux bases du langage avant de s'aventurer progressivement vers des notions plus avan-

cées. Ce livre est donc idéal pour commencer l'apprentissage de PHP ! D'un autre côté les utilisateurs expérimentés y trouveront quant à eux une source de perfectionnement complète. Cet ouvrage décortique à travers 30 chapitres chaque problématique du développement web avec PHP. Les expressions régulières, les systèmes de cache et la gestion des images sont quelques-uns des nombreux sujets abordés. Des cas d'application permettent de mettre en pratique chacune des thématiques déjà illustrées par de nombreux exemples et schémas. En plus de présenter les dernières nouveautés, cette 5e édition propose également un guide de migration vers PHP 5, passage désormais obligé depuis l'arrêt récent du support de PHP 4. Un tutoriel d'introduction au framework Symfony est aussi au rendez-vous. Enfin, une multitude de logiciels PHP sont présentés afin de vous guider dans le choix de tels outils.

■ Vincent Brouté, <http://www.depannetonpc.net>

livre du mois

de la couleur, les objets d'interface, les couches de code, XAML, audio / vidéo, les données, Linq, accès distant, templates, interaction avec HTML, tout y passe, ou presque. On regrette cependant que la partie développement pur soit reléguée en annexe car cela aurait mérité un traitement en profondeur, tout comme la sécurité, le déploiement ou encore Silverlight Streaming. Une bonne introduction même si le développeur avancé cherchera ailleurs.

Adobe Flex 3



Difficulté : ***
 Editeur : Adobe / Pearson
 Auteur : collectif
 Prix : 49 €

Flex 3 offre une plate-forme d'exécution et de développement très complète avec maintes possibilités. Encore faut-il la maîtriser ! Ce livre est une véritable formation de A à Z sur Flex, la technologie, les outils. En 21 leçons, vous allez pas à pas : concevoir l'interface, gérer les événements, connaître MXML, gérer les données, intégrer des services web, attaquer les services serveur, etc. Toujours très clair,

l'ouvrage est une aide précieuse pour mieux appréhender Flex. Pour le développeur web, ce titre s'impose. Il inclut un CD Mac et PC.

The Productive Programmer



Difficulté : ***
 Editeur : O'Reilly
 Auteur : Neal Ford
 Prix : 32 €

Qu'est-ce qu'un développeur productif ? Déjà un développeur peut-il être productif ?

La réponse est oui et Neal tente de répondre à cette question durant tout le livre. La productivité passe par la méthode, les outils, les petits riens qui au quotidien gênent ou améliorent le travail. C'est pour cela que Neal découpe son ouvrage en deux parties : les mécanismes et la pratique. On oscille entre "c'est évident pourquoi n'y ai-je pas pensé" et "je n'y comprends rien à la philosophie grecque !". Certains chapitres nécessitent plusieurs lectures pour comprendre la finalité et ce que Neal veut dire, car on ne voit pas forcément le lien entre développeur et philosophe de la Grèce classique ! Au final, un excellent ouvrage que nous vous conseillons de lire. En Anglais.

Il s'en est passé des choses en 2008...

❄️ **Les députés migrent sous Linux**

❄️ **Près de 30 000 postes sous OBM au Ministère de l'Intérieur**

❄️ **Plus de 1 200 utilisateurs OpenOffice.org chez l-CDC**

❄️ **L'INSERM choisit OBM pour la puissance de son agenda partagé**

❄️ **Weishardt modernise sa messagerie avec OBM**

❄️ **France 24 choisit Drupal pour la refonte de son site Internet**

... et vous n'avez encore rien vu !

LINAGORA vous souhaite de très belles fêtes de fin d'année.
Merci à nos clients, à nos collaborateurs et à nos partenaires
pour leur confiance et leur travail.

www.linagora.com

LINAGORA

Votre potentiel, notre passion.™
Microsoft®

Transformez les données de votre
entreprise en énergie nouvelle.

Découvrez Microsoft® SQL Server 2008®. Maîtrisez la puissance de vos données. Le nombre de formats de données que vous avez à traiter ainsi que leur volume ont explosé. Grâce au nouveau SQL Server 2008, domptez la puissance inexploitée de l'ensemble de ces données. Vous pouvez les intégrer, les gérer efficacement, et même les partager avec vos utilisateurs finaux, comme vous n'auriez jamais imaginé pouvoir le faire. Pour en savoir plus, rendez-vous sur : www.microsoft.com/france/sql

