LE MAGAZINE DU DÉVELOPPEMENT

Programmez !

Mensuel - Mai 2006 - N°86 • 5,95 €

JAVA 6 - Java EE 5
La renaissance ?

P2P
Une nouvelle génération de réseau

3 technologies expliquées : BitTorrent, Java, .Net

www.programmez.com
Tout nouveau, tout beau !

.NET Tester vos applications avec NUnit
Améliorer et sécuriser les Web Services
JAVA Utiliser la classe singleton
C++ GDI+ : une API graphique surpuissante
TECHNO Concevoir un projet RFID
SÉCURITÉ Les nouveautés d'IE7

PHP Découvrir le framework Zend

Printed in France - Imprimé en France -
BELGIQUE 6,45 € - SUISSE 12 FS -
LUXEMBOURG 6,45 € - Canada 8,95 $ CAN

M 04319 - 86 - F: 5,95 €

ANASKA LE SPECIALISTE FRANCAIS DE LA FORMATION OPEN SOURCE

Formation Expert Sécurité Linux

Maîtriser toutes les tâches d'administration et la sécurité d'un serveur Linux

AU PROGRAMME :

* Mise en place de stratégies de sauvegarde
* Comprendre et analyser les logs
* Gestion des performances
* Mise en place de stratégies de sécurité
* Gestion des utilisateurs
* Comprendre et prévenir les principales attaques

Prochaines sessions

Paris
26/06, 11/09, 16/10, 11/12

Strasbourg, Lyon, Nantes
16/10, 11/12

anaska
Conseil & formation en logiciels libres

Nos parcours de formation Linux

Linux devient incontournable sur le marché des systèmes d'exploitation.
Nos formations vous accompagnent de la prise en main à des niveaux d'expertise poussés.

Parcours administrateur

Linux les commandes | Linux Administration 1 | Linux Administration 2 | Linux Sécurité

Parcours connexes

Samba Partage de fichiers | Perl

Informations : 01 45 28 09 82
http://www.anaska.com

Anaska adhère à l'Association des Sociétés de Services en Logiciels Libres

sommaire

n°86

ACTUS

Agenda 3
L'actualité en bref 6

TESTS

REALbasic : l'autre basic 11
BackBase 12

SÉCURITÉ

Internet Explorer 7 : le renouveau 13

CAS ENTREPRISE

Feriscope, l'Intranet de RFF, revu et refondu 16

FOCUS

P2P : Créer un réseau nouvelle génération

Sécurité des réseaux P2P : fantasme ou réalité ? 19
BitTorrent au service de l'industrie et des professionnels ! 20
Le framework P2P JXTA 23
P2P sous . Net avec les sockets 25

DOSSIER

Java 6 et Java EE 5 : la renaissance ?

Java 2006 : un agenda chargé 29
La stratégie de Sun 32
Avis d'experts 35
Java EE 5 : vers la simplification 36
Développez des EJB3 avec EasyBeans 39
Java 6, le rapprochement de Java et du système d'exploitation 43
Java Mustang, une JVM plus proche de votre système d'exploitation 46

CARRIÈRE

Java, première compétence métier 50

TECHNOLOGIE

Le contrôle qualité des applications Java 54
Comment développer une solution RFID 56

PRATIQUE

Zend Framework : un outil prometteur 62
Tests unitaires avec Nunit 66
Sécuriser un service Web avec les WSE 3.0 70
Le Singleton, un objet unique en son genre 74
Se familiariser avec GDI 77

TEMPS LIBRE

Ludique 80
Livres 82

Agenda

MAI

Du 03 au 04 mai 2006
Paris La Villette,
Cité des sciences et de l'industrie
Webmasters Expo,
www.webmasters-expo.com

Les 10 et 11 mai
Paris La Défense, CNIT
Communications over IP
Conférences pratiques autour des technologies de communication sur IP
www.salon-coip.com

Le 16 mai 2006
Paris 3e, CNAM - 292, rue Saint Martin
Journée **Neptune 2006**
Processus de développement logiciel, basé sur les modèles : démarche et outillage.
Inscriptions :
genie-logiciel@wanadoo.fr
http://neptune.irit.fr

Les 17 et 18 mai 2006
Paris La Défense, CNIT
Online 2006
Le rendez-vous des professionnels du Marketing, de la Publicité et de la Relation Client sur Internet.
www.online-2006.fr

Du 30 mai au 1er juin 2006
Paris- Expo, Porte de Versailles
SAPPHIRE 2006
L'événement annuel européen dédié aux clients SAP
www.sapsapphire.com/emea2006

ETRANGER

Du 16 au 19 mai 2006.
Etats-Unis, Moscone Center, San Francisco
JavaOne Conference.
Conférence mondiale organisée par Sun Microsystems
www.java.sun.com/javaone/sf

Du 13 au 17 août 2006
(inscriptions ouvertes)
Etats-Unis, Orlando
FileMaker Developer Conference, sur le thème : porter les solutions FileMaker sur le Web
www.filemaker.com/devcon

FORUM JAVA

Dialoguez avec les experts Java
« Le nouveau Java, toutes les questions que vous vous posez »

Avec nos invités experts :
- **Hervé Crespel,** Président du Club des Utilisateurs de Java
- **Thibaut RÉGNIER,** Secrétaire Général du Club des Utilisateurs de Java
- **Alexis Moussine-Pouchkine,** architecte Java, Sun France
- Animateur :

Frédéric Mazué, Programmez

www.programmez.com

ETES-VOUS A LA RECHERCHE DU MEILLEUR MOYEN DE DEVELOPPER DES APPLICATIONS MULTIPLATEFORMES PERFORMANTES?

Plus besoin de choisir entre performance et portabilité – les deux sont possibles, grâce à Qt et son environnement de développement C++ multiplateforme. Avec Qt vous pouvez dorénavant développer des applications natives sur Windows, Unix/Linux et Mac OS X à partir d'un seul et unique code source. Sans couche d'émulation ni machine virtuelle, Qt vous permet de développer des applications hautement performantes sur toutes les plateformes.

Qt vous permet aussi d'accroître votre productivité. Plus besoin d'écrire et d'entretenir deux codes sources, vous pouvez dorénavant laisser libre cours à votre créativité, Qt et son API simple d'utilisation se chargent des aspects non portables de votre application. Pas étonnant que des sociétés telles que Google®, Adobe®, Synopsys® et Perforce® aient adopté Qt pour leurs développements majeurs.

Pour en savoir plus ou pour télécharger une version d'évaluation, rendez-vous sur notre site: www.trolltech.com/lookhere

Qt® BY TROLLTECH

CODE LESS. CREATE MORE.

Trolltech, Qt, and the Qt logo are registered trademarks of Trolltech AS in Norway and/or other countries worldwide. All other trademarks are property of their respective owners.

TOUT NOUVEAU TOUT BEAU

Programmez.com change !

LE MAGAZINE DU DÉVELOPPEMENT

Programmez !

Rédaction : redaction@programmez.com

Directeur de la Rédaction : Jean Kaminsky.

Rédacteur en Chef : François Tonic.

Ont collaboré : F. Benoit, F.Boudinet, J.Chable, D.Decroos, G. Delamarre, J.M. Fontaine, C.P. de Geyer, A. Kerhuel, X. Leclercq, P. Lemberger, J.M. Maman, F. Mazué, A. Meraoumia, C.Padych, C.Peyrusse, J.Renard, J. Viale.

Maquette : PLB Communication.

Dessin : M. Piedoue.

Crédit photo couverture : ©Larry Williams/Zefa/CORBIS

Publicité : régie publicitaire : ECC, Stéphane Kaminsky - *pour la publicité uniquement :* Tél. : 01 56 31 21 54 - Fax : 01 56 31 21 59 - publicite@programmez.com.

Editeur : Go-02 sarl, 6 rue Bezout - 75014 Paris
Dépôt légal : 2e trimestre 2006
Commission paritaire : 0707K78366 - ISSN : 1627-0908
Imprimeur : ETC - 76198 Yvetot

Directeur de la publication : Jean-Claude Vaudecrane.

L'édition destinée aux abonnés comporte un cd rom et un encart rédactionnel

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 - Abonnement (magazine seul) : 1 an - 11 numéros - France métropolitaine : 45 € (295,18 FF TTC) - CEE et Suisse : 51,83 € - Algérie, Maroc, Tunisie : 55,95 € - Canada : 64,33 € - Tom : 79,61 € - Dom : 62,84 € - Autres pays : nous consulter.

Edito

Et si on réinventait le logiciel ?

À quoi pourrait ressembler le logiciel de demain ? Nous en avons déjà de nombreux exemples avec les applications Ajax et tout ce qui tourne autour du web 2. Mais il s'agit d'aller plus loin encore : une application composite, constituée de petits modules correspondant à une ou plusieurs fonctions que l'on agglomère dans un conteneur.

Il est possible de faire un parallèle avec les générations de téléphones portables :
- 1G : le site web ou l'application desktop,
- 2G : l'interactivité, le multimédia sur le site Web, avec des notions de clients lourds, riches, légers,
- 2,5G : éclosion du Web 2 et de son langage phare AJAX apportant l'interactivité supérieure, une interface plus riche, plus ergonomique, un web plus collaboratif.

La 3G, pourrait correspondre à la rupture déjà engagée avec le Web 2 et AJAX. On porte les applications desktop sur le navigateur web ou toute application conteneur, en les découpant, en les proposant en accès gratuit et libre ou payant. L'étape suivante consiste à se passer du système d'exploitation tel qu'on le connaît. Le navigateur web (ou autre) fera office de système d'exploitation...

Le temps pluvieux aurait-il grippé mon cerveau ? Pas plus que d'habitude... Aujourd'hui, il est clair que le modèle applicatif, que ce soit en desktop, en site web, arrive à la fin d'un cycle. Google, Yahoo !, Microsoft, IBM, et d'autres, ont compris que le virage technologique était déjà en cours. L'application monolithique coûte cher à acheter, à maintenir, à développer. Si l'open source résout le coût, il ne résout pas la complexité. Mais il constitue un pilier de l'application de demain. Imaginez un Open Office ou un Microsoft Office que l'on « construirait » soi-même en activant ou non les différentes fonctions. Imaginez le bureau virtuel (qui existe déjà) disponible sur son navigateur permettant l'accès à toutes les applications, son courrier, son agenda, etc. Ajax montre la voie de la segmentation, XUL et XAML de la nouvelle notion d'interface, le Web de l'esprit nouveau, et l'architecture SOA de l'interaction entre les différents composants.

Bien entendu, Java et le P2P font partie des évolutions applicatives. Pourquoi le P2P ? Cette technologie facilite la conception d'un réseau pour échanger données et fichiers, communiquer entre utilisateurs. Dans notre dossier, vous verrez que le P2P mérite votre attention et qu'au-delà des usages répréhensibles, il permet beaucoup de choses. Notre grand dossier du mois concerne le futur de Java avec Java 6 et Java EE 5. Nous ferons aussi une petite excursion sur Java 7...

François Tonic

PROCHAIN NUMÉRO n°87 - Juin 2006 - Parution le 31 Mai 2006

Créer son environnement idéal

Éclipse

DOSSIER

Comment configurer l'IDE ?
Comment choisir les plug-ins ?

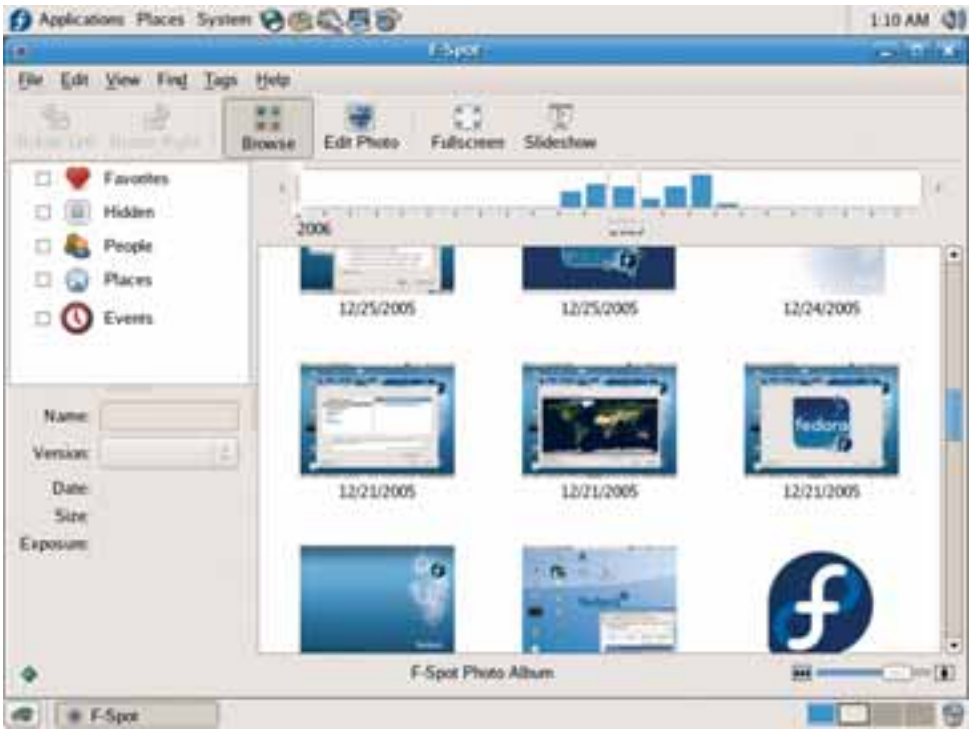
SOA

FOCUS

Le système d'information est en pleine mutation. Les architectures orientées service deviennent le nouveau maître-mot.

Red Hat rachète JBoss…

La rumeur d'un rachat de JBoss circule depuis quelque temps déjà dans les milieux de l' informatique. C'est finalement Red Hat qui emporte le morceau. Par ce rachat, Red Hat se dote d'une offre logicielle serveur et de middleware complète. Red Hat annonce avant tout sa volonté de se positionner sur le très prometteur marché des architectures SOA qui nécessite une infrastructure logicielle fiable et performante, et à bas prix pour l'utilisateur, même si Red Hat paie au prix fort l'éditeur open source : environ 350 millions ! Gartner estime à 6,5 milliards de dollars le marché de l'intégration d'application, du middleware et du portail. La concurrence est déjà très forte, notamment avec les éditeurs commerciaux comme IBM, SoftwareAG, BEA, Microsoft…



Quelques jours avant cette annonce, JBoss présentait deux projets importants pour sa suite middleware : JBoss Messaging et JBoss Web. JBoss Messaging constitue le noyau de messagerie pour les architectures SOA et les bus ESB. Il prend en charge Java Message Service. Et constitue un module standard de JBoss 5.0, prévu pour 2006. JBoss Web est un serveur web reposant sur Tomcat, compatible avec les applications JSP, ASP.Net, PHP, CGI. L'annonce la plus importante remonte à début avril avec JBoss Transaction, un moteur transactionnel basé sur la technologie Arjuna et HP, racheté par JBoss en 2005. Dans la foulée, l'éditeur annonçait JBoss jBPM 3.1, un outil de gestion des processus métiers en Java. Toutes ces annonces se faisaient autour d'un thème commun : SOA.

…et arrête la fondation Fedora !

Un peu à la surprise générale, Red Hat annonçait début avril le retrait de son soutien à la fondation Fedora, initiée en juin 2005. L'éditeur réorganise donc, en dehors de toute fondation soutenue financièrement par Red Hat, Fedora Project pour redonner une direction claire au projet. Tout d'abord, Fedora retrouve un bureau incluant des leaders de la communauté : 5 venant de Red Hat, 4 de la communauté Fedora. Ce bureau prendra en charge les décisions opérationnelles, les décisions sur le budget et les choix stratégiques. En quelque sorte, Fedora revient dans le giron Red Hat, même si l'entité demeure à l'extérieur. Pour preuve, un « patron » payé par Red Hat aura un droit de veto sur les décisions du bureau. Ce patron sera Max Spevack, déjà très présent à Fedora. Reste à éclaircir rapidement le quotidien du projet et les choix stratégiques. L'échec de Fondation est de l'aveu même de l'éditeur dû à une trop grande ambition dès le départ. À l'origine, la fondation était vue comme un entrepôt à brevets logiciels pour la communauté Open Source. Red Hat a depuis privilégié le travail autour des brevets, avec l'Open Invention Network qui collecte des brevets pouvant être librement utilisés. Si la communauté souhaitait un Fedora plus indépendant de Red Hat, car Fedora est une émanation de Red Hat et est à la base de ses distributions, l'éditeur pouvait difficilement laisser agir Fedora en toute indépendance. C'est toute la difficulté d'éditeurs comme Red Hat qui composent des solutions commerciales sur une fondation open source / libre. Le plus délicat est de trouver le bon équilibre entre les besoins de l'entreprise pour satisfaire les clients et le marché, et la communauté, pour qu'elle ne soit pas « dépossédée » du projet…

Fin mars, Fedora Core 5 était officiellement disponible en téléchargement, après quelques mois de retard. Elle inclut toutes les dernières évolutions des packages open source et du noyau Linux. Surtout, la distribution possède en standard la plate-forme logicielle et de développement Mono, le « clone » de .Net racheté par Novell.

Site Fedora :
http://fedora.redhat.com/

DÉMARRAGE

Apple ouvre la porte à Windows

L'annonce du logiciel Boot Camp par Apple a fait l'effet d'une petite bombe. Depuis le passage aux processeurs Intel par la Pomme, de nombreux bidouilleurs cherchaient à faire fonctionner MacOS X sur un vulgaire PC et Windows sur un Mac Intel ! Boot Camp comble là les désirs de nombreux utilisateurs : faire fonctionner sur un Mac Intel, MacOS X et Windows, sans avoir besoin de deux machines. Boot Camp permet en effet de booter (et d'installer) Windows (XP) sur les nouveaux Mac. Surtout, Apple annonce que l'outil sera intégré dans le prochain MacOS X, Leopard dont aucun détail n'est actuellement connu. Les retards de Windows Vista laissent une certaine marge de manœuvre à Apple et à son MacOS X. Boot Camp oblige à redémarrer le Mac pour changer de système et n'est pas donc une solution de virtualisation.

Site : www.apple.com

ALTOVA
mapforce
2006
Améliorez vos
Services Web
Associez-vous à MapForce® 2006 et définissez
des Services Web sans écrire de code
Nouveautés dans la version 2006:
• Implémentation drag and drop de Services Web
• Parsing et intégration avancés de fichiers texte
• Génération de code du projet entier
• Intégration dans vos applications vers OLE/ActiveX
Wanted:
C++ Developers
EDI
XML
DB
Legacy
Altova MapForce, l'outil réputé pour son intégration
simple de XML, des bases de données, des fichiers
texte et de l'EDI, vous permet désormais d'implémenter
des Services Web de manière visuelle. Il vous suffit
d'établir des liens entre votre source et votre cible
d'informations, et de leur associer des fonctions de
traitement de données. MapForce convertit les données
en un clin d'oeuil et autogénère un code de mapping
personnalisé en XSLT 1.0/2.0, XQuery, Java, C++ et
C# pour votre utilisation dans vos applications
d'intégration de données ou de Services Web.
Apportez un sens à vos données.
Téléchargez MapForce® 2006 dès
aujourd'hui: www.altova.com
ALTOVA

QUALITÉ

OptimalJ et Eclipse

L'éditeur Compuware vient d'annoncer la sortie d'OptimalJ 4.1 se basant sur Eclipse, nouvelle preuve qu'Eclipse constitue la plate-forme technique incontournable du monde Java. Eclipse hérite ainsi d'un puissant modeleur MDA, un des plus reconnus du marché. À noter qu'OptimalJ supporte aussi NetBeans. Sur la partie UML, l'outil prend en compte MagicDraw de No Magic Inc et on peut maintenant afficher les designs patterns en diagrammes UML 2. Trois éditions sont disponibles : Developper, Professional et Architecture.

Site : www.compuware.com

INDUSTRIE

National Instrument complète LabView

L'éditeur NI, spécialisé dans le matériel et les environnements de mesures, complète son outil phare, LabView, en sortant une nouvelle version de LabView Control Design Toolkit. Il s'agit d'un ensemble d'outils de fonctions mathématiques pour l'analyse, le design et l'implémentation de systèmes de contrôle. Il est capable de fonctionner en temps réel et inclut des filtres Kalman. NI, en association avec Analog Devices, sort une extension à LabView : LabView Embedded Module for ADI Blackfin Processors. Ce module permet à LabView de cibler les processeurs consommation Blackfin utilisés dans l'embarqué. De plus, ce module permet, même sans connaissance du développement embarqué, d'analyser et de résoudre les problèmes des applications.

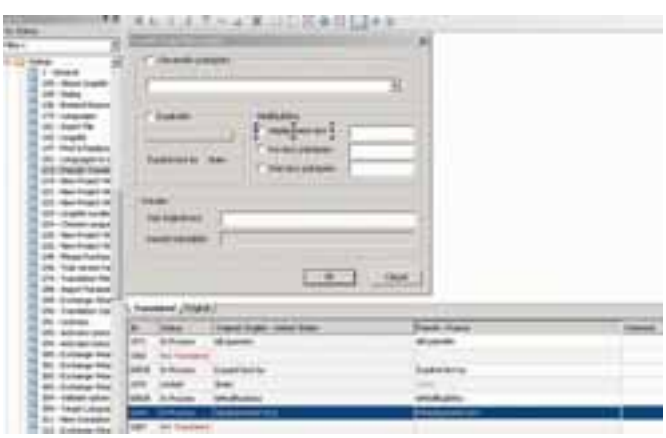
Site : www.ni.com

LOCALISATION

Localisation facilitée avec Localizer 4.5

La localisation des applications demeure une tâche fastidieuse et longue pour les développeurs. L'éditeur Lingobit propose la nouvelle version de son outil Localizer prenant en charge les applications .Net, Windows, Java. .Net 2 fait son apparition, ainsi que le support de Delphi et des applications codées en C++ Builder. L'outil inclut des rédacteurs et des traducteurs pour faciliter le travail. Plus de 180 langues sont supportées.

Site : www.lingobit.com



SOCIÉTÉ

Wind River rachète !

L'éditeur Wind River vient d'acquérir la société Interpeak, spécialisée dans la sécurité, les réseaux et la mobilité. Wind River parle de convergence de vue entre les deux sociétés sur le marché des équipements connectés. Wind River intégrera dans VxWorks et Linux des piles sécurisées de middleware de mise en réseau, un élément vital dans la connexion des matériels. L'éditeur annonce aussi une initiative ayant pour but la standardisation et l'adoption de l'architecture multicore dans les équipements connectés. Arm, Broadcom et Freescale sont partenaires de l'initiative.

IBM

Annonces Eclipse…

IBM n'avait jamais réellement perdu sa place dans le projet Eclipse même si aujourd'hui, de nombreux autres éditeurs participent aux divers projets associés. Initiateur du projet il y a plus de 4 ans, en 2004, l'éditeur avait activement participé à la création de la fondation Eclipse. IBM a annoncé fin mars de nouvelles initiatives autour d'Eclipse :

- une compétition mondiale « **Eclipse Innovation Award Program** » : vise à encourager l'usage de l'open source et de la création d'outils ouverts dans les universités et chez les chercheurs.
- **Rational Data Architect** : ce nouvel outil orienté données d'entreprise doit aider les architectes données à comprendre la structure, l'intégration, la gestion des données. Doit aussi aider à la fédération de grands volumes de données.
- **Workplace Forms Designer 2.6** : basé sur Eclipse 3.1, cet outil permet de créer des formulaires XML (e-forms). La sortie est prévue courant juin prochain.
- Aide : comment créer et mettre en place rapidement une aide utilisateur dans une application créée avec Eclipse ? IBM lance le projet **Interface Help System built on Eclipse**. L'outil est multi lingue et multi-plate-forme et l'aide générée peut s'utiliser en ligne, en local, sur un CD, etc.

…SOA et DB2

Au-delà d'Eclipse, IBM n'oublie pas les autres marchés et notamment les architectures de services de type SOA. La dernière annonce en date est : **SOA Gouvernance**, un portfolio pour aider à gérer et à prendre les décisions d'architecture et de gestion. Dans une architecture SOA complexe, la gestion devient une problématique vitale. Au-delà de 50 services, on estime qu'il est nécessaire d'utiliser un outil de gestion, de gouvernance.

Enfin, attendue depuis plusieurs semaines, IBM vient de lancer la version d'évaluation de **DB2 Viper** ! Viper, la prochaine version de DB2 prévue pour la mi-2006, prend maintenant en compte les architectures SOA en améliorant l'accès et la gestion des données dans un cadre SOA (avec un accès simplifié et sécurisé aux données). DB2 Viper gère nativement les données XML et relationnelles.

Site : www.ibm.com/db2/viper

FRAMEWORK

LEONARDI nouvelle version

L'éditeur français Lyria vient d'annoncer la version 3.2 de sa librairie d'interface LEONARDI. Cette version intègre désormais le support d'AJAX et des fonctions afin d'automatiser les tests. Sur les tests, il est possible d'enregistrer les actions de l'application, d'y intégrer des contrôles et ainsi de créer des tests tout au long de sessions. Ce sont des tests fonctionnels, s'assurant ainsi de la non régression de l'application. Il est possible d'intégrer des points d'arrêts ou même d'arrêter la session de tests, varier la vitesse d'exécution, etc. Ces scénarios peuvent ensuite alimenter des outils tels que Junit. Sur le support d'Ajax, il s'agit avant tout d'offrir une meilleure interactivité, une interface encore plus riche. LEONARDI est disponible à partir de 10 000 euros.

Site : www.lyria.com

SALON

Salon RTS sur sa lancée

Avec la mort du SETI, grand lieu de rassemblement de différents salons informatique, les salons parisiens avaient eu du mal, en 2005, à trouver leur place. L'édition 2006 du salon RTS s'est tenue au CNIT à Paris La Défense et a redoré l'image du salon. La majorité des nouveautés du salon RTS se concentraient sur le matériel. On notait la présence des acteurs classiques du marché : QNX, Wind River, Lynuxworks. Microsoft était présent par son stand et surtout, un grand atelier autour de Windows CE et XP embarqué. Côté application critique et embarquée, on pouvait aller voir JamaicaVM qui permet d'utiliser Java en temps réel avec une JVM adaptée aux contraintes critiques et de l'embarqué. Aujourd'hui, Java 1.4 est supporté. Java 5 est planifié pour 2007.

Un petit salon annexe, mais se fondant dans RTS et Display, est promis à un très bel avenir : le salon M2M (Machine to Machine). Il s'agit de pouvoir faire communiquer directement deux appareils sans que l'utilisateur soit l'intermédiaire, ou qu'un ordinateur assiste à la communication. Le M2M concerne de très nombreux domaines : téléphonie, réseau, sécurité, santé, etc. Les applications possibles sont immenses. Pour preuve, en 2007, il est envisagé de créer un espace distinct pour le M2M ! Enfin, sur le salon Display, on pouvait voir un écran un peu particulier de la société Newsight. Il s'agit d'une gamme d'écran 3D ne nécessitant aucune lunette spéciale ! La mise en relief de la 3D est très impressionnante et réaliste !

3.248 heures pour développer le code…
184 pour trouver ce sacré bug!
142 heures de réunions…
108 pizzas!
14 week-ends à l'eau!
11 nuits blanches…
Protégez tout d'un seul appel!
HASP
SOFTWARE DRM
En choisissant les solutions de protection de nouvelle génération HASP, vous offrez – en une seule opération – une protection ultime à vos créations logicielles tout en bénéficiant d'un environnement simple et transparent pour gérer les options de vos licences.
1 Développement
2 Protection unique
3 Multi-distribution
Demandez votre kit Développeur, ainsi que son guide technique et une clé de démonstration sur www.aladdin.com/hasp
Aladdin
SECURING THE GLOBAL VILLAGE
Amérique du Nord: 1-800-562-2543 International: +972-3-636-2222 UK: +44-1753-622266
Allemagne: +49-89-894-221-0 Benelux: +31-30-688-0800 France: +33-1-41-37-70-30 Espagne: +34-91-375-99-00
Israël: +972-3-636-2222 Asie-Pacifique: +852-21-66-8605 Japon: +81-426-60-7191

AGL

Nat Systems remet à jour sa gamme

L'éditeur français a profité du printemps pour mettre au goût du jour ses gammes. Tout d'abord, c'est la sortie de la 5.0 de son outil de développement simplifié NS-DK. L'interface et l'ergonomie de cet atelier ont été revues pour améliorer la prise en main et l'espace de travail. Le langage de programmation possède maintenant des API de trace et de collection. Sur les données, NS-DK ajoute la compatibilité avec Oracle 10 G release 2 et SQL Server 2005. De plus, la 64-bit est maintenant disponible sur la version serveur. Dans la gamme d'outils de l'éditeur, il y a aussi NatJxt (jxt pour Java XML Terminal). Il s'agit de pouvoir développer des applications métiers 100 % Java – Java EE. L'application métier est mise en production en mode JXT, une applet générique, ou en mode AJAX. Le tout reposant sur le framework de présentation JxT Client Léger Riche. L'IDE NatJtx s'appuie sur Eclipse.

Site : http://www.natsystem.fr

OUTILS

JDEO génère les applications

L'éditeur JDEO annonce la v2.1 de son logiciel de génération d'applications web, reposant sur la plate-forme Java EE. Son fonctionnement est simple : le concepteur définit des composants applicatifs, les positionne au niveau de son interface et les intègre au sein de fonctionnalités, sans nécessiter aucun développement. JDEO propose des composants de présentation et de « données » dont le cycle de vie est géré par les séquences ou fonctionnalités. Le parallèle avec les composants techniques J2EE et leurs cycles de vie gérés par les conteneurs est facile. Ce niveau d'abstraction repose sur le concept de Programmation Orientée Définition. La séparation entre la définition de l'application en terme de fonctionnalité et la technologie, permet de faire évoluer l'un de ses aspects sans impacter l'autre, d'où un gain en termes de charges. JDEO gère la couche présentation des applications, ce qui exige un niveau d'interopérabilité très souple avec les bases de données, les composants ou services développés par ses clients.

Site : www.JDEO.com

COMPOSANTS

PaseoExpress fait le 33

PaseoExpress de l'éditeur Paseosoft est un outil pour créer rapidement des scripts, via 33 composants .net, facilement paramétrables et personnalisables. On peut graphiquement définir les enchaînements, automatiser des tâches, vérifier le bon fonctionnement d'une application, etc. L'outil joue sur la simplicité de conception et sa souplesse d'utilisation.

Site : http://www.paseosoft.fr

PROCESS

Telelogic et Eclipse

L'éditeur Telelogic s'engage fermement dans Eclipse en participant à l'Eclipse Process Framework (EPF). L'éditeur apporte avec lui une large bibliothèque de bonnes pratiques sur les exigences, MDA, la gestion des changements. EPF possède aussi un volet outil, avec EPF Composer, pour autoriser et publier des processus, il sera fourni avec des exemples.

SGBD

MySQL flirte avec Eclipse

L'éditeur MySQL AB vient de rejoindre la fondation Eclipse. L'éditeur de la base de données MySQL collaborera activement au projet Data Tools Platform et travaillera de concert avec Zend (et d'autres) sur le projet Eclipse PHP IDE. Le projet vient tout juste d'être créé, il faut encore le définir et le planifier. Il intégrera un PHP IDE Core (infrastructure de l'IDE), PHP IDE UI (pour l'interface), PHP IDE Debug pour la partie debug et tests. Le but est de fournir rapidement une version prototype et une version fonctionnelle dans les plus brefs délais.

Site : http://www.eclipse.org/proposals/php-ide/

SGBD

Alliance MySQL tout azimut

MySQL a lui aussi profité du printemps pour annoncer la signature d'alliances avec IBM et EMC autour du programme MySQL Network Certified Partner. Ce programme vise à fournir des outils et services aux utilisateurs de MySQL. L'éditeur libre recommande d'ailleurs IBM WebSphere DataStage pour faire de l'ETL ou du datawarehouse. De plus, MySQL fonctionne sur POWER et les principales gammes serveurs d'IBM. EMC propose pour MySQL des outils de stockage, de protection et de gestion des données avec des mécanismes de virtualisation et de réplication.

SÉCURITÉ

F-Secure protège les mobiles

L'insécurité étant la nouvelle préoccupation des terminaux mobiles, F-Secure, avec Orange, lance un outil antiviral pour mobile. Orange proposera automatiquement F-Secure pour mobile. Il sera accessible via le portail Orange World. L'utilisateur bénéficiera d'une actualisation automatique de l'antivirus. Une phase pilote avait eu lieu en 2005. Le logiciel est utilisable pour test durant un mois, après il devient payant (paiement mensuel).

REALbasic 2006 : l'autre basic !

Le langage Basic n'est pas mort. Il existe de nombreuses déclinaisons multi-plates-formes ou non. REALbasic, de l'éditeur REAL Software, est désormais disponible en version 2006 et en Français.

REALbasic est un outil de développement en langage basic avec des notions objets. Il est doté d'un designer graphique pour construire l'interface et permet de compiler pour MacOS X, Windows et Linux, à partir du même code source ! L'outil prend en charge les spécificités de chaque plate-forme, notamment pour l'interface. Il suffit de configurer les options de build. L'éditeur ne s'embarrasse pas avec les fonctions avancées ou les options dans tous les coins. L'IDE va à l'essentiel, avec une interface plutôt ergonomique et fonctionnelle. On accède rapidement à la partie code et designer. Le choix des objets graphiques est très varié et couvre l'essentiel des besoins. On dispose d'une gestion de projet simple mais adaptée. À noter que REALbasic est multi projet, à chaque projet, il ouvre une nouvelle fenêtre environnement. La version 2006 inclut de nombreuses nouveautés : un contrôle HTML, un moteur de données basé sur SQLLite, le support de Spotlight d'Apple, un meilleur support de Linux. Dans l'éditeur, on dispose de la complétion de code, même si celui-ci ne va pas aussi loin qu'Eclipse ou NetBeans. Sur le code, il n'y a pas de détection à la volée des erreurs. L'outil, en téléchargeant un module externe, est capable de reprendre du code VB. Il faut passer le code VB dans le VB Project Converter. Il suffit de récupérer le projet VB, d'ouvrir le fichier projet, l'outil détecte automatiquement l'ensemble des fichiers (code, forms...) liés. La conversion se réalise dans un fichier XML que l'IDE charge et transforme en projet REALbasic. Il affiche alors le code converti et les fichiers interfaces. Mais vous aurez tout de même un travail de vérification et de correction à faire sur l'interface (notamment sur le positionnement des objets) et le code. Plus un code sera complexe (ou mal écrit), plus il sera nécessaire de le vérifier. Cependant, le convertisseur demeure une bonne solution pour aller vite et mâcher le travail. VB.Net n'est pas pris en charge. Vous aurez le choix entre l'édition standard et professionnelle. La standard n'inclut pas la compilation multi système, le support étendu des SGBD, les connexions SSL ou encore le debug distant. Le support du Mac Intel est planifié.

■ **François Tonic**

Fiche technique

REALbasic 2006
Editeur : REAL Software
Site : www.realsoftware.com/fr/
Prix : à partir de 85 euros (version standard)
Configuration : Linux, Windows ou MacOS X, 512 Mo de mémoire minimum

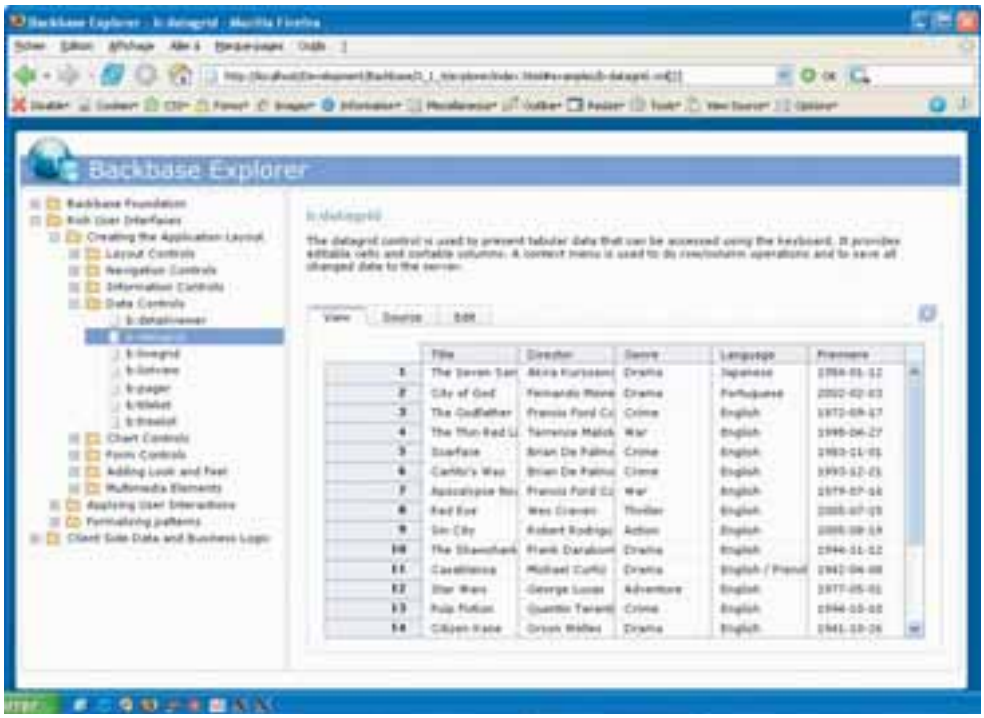
Les +

- Une interface claire
- La compilation croisée
- Le convertisseur VB
- Le designer

Les -

- Fonctions avancées absentes
- Pas de vérification à la volée du code
- Des modules non intégrés

Backbase : un framework AJAX surpuissant !



Depuis quelques mois, voire plus, les concepts Web 2.0, à commencer par Ajax et ses variantes, envahissent le monde Internet. Et indiscutablement, le déploiement massif d'Ajax dans les sites et applications web constitue une révolution. Malheureusement, si le résultat est très impressionnant, le développement de fonctionnalités Ajax est souvent très fastidieux.

BackBase est probablement, à l'heure actuelle, le framework Ajax à la fois le plus complet et le plus abouti. Bâti autour d'un concept relativement original, reposant sur un dialecte XML créé pour l'occasion, il permet de réaliser vite et bien des applications web dotées d'interfaces remarquables. Ce dialecte, le BXML, permet de décrire la quasi totalité des contrôles graphiques que propose le framework. Dans certains cas, il suffit simplement d'appliquer un style particulier, défini lui aussi par le framework, pour changer totalement le comportement d'un élément standard, comme une DIV ou un tableau (Ex : appliquer un style spécifique à une DIV permet de la déplacer ensuite). Ce dialecte est interprété par le navigateur client, grâce à un run-time JavaScript, chargé de manière transparente lors de la première consultation, et ensuite mis en cache.

Installation de Backbase

L'installation du framework est parfaitement triviale, puisqu'il suffit de télécharger une archive et d'en extraire le contenu dans l'espace public (Document Root) de votre serveur web. Deux dossiers principaux seront créés : l'un pour le développement, l'autre pour la production. Une fois votre application écrite et testée, il ne reste plus qu'à exposer en ligne la partie production !

Une utilisation simplifiée

Une fois installée, la plate-forme permet de découvrir de nombreux exemples, grâce au Backbase Explorer. Ce qui constitue la première occasion de se laisser impressionner par le nombre de contrôles graphiques proposés par Backbase. Des menus dynamiques aux listes triables, y compris des listes « live », permettant de créer une liste de plusieurs milliers d'éléments sans problème, car seules les lignes affichées sont chargées depuis le serveur, sans oublier le glisser/déplacer, un système de fenêtrage, des boîtes à onglets, etc. Impossible les lister tous !

Et si cela ne vous suffisait pas, alors nous vous invitons à tester également les « starter kits », qui vous montreront des applications types de l'utilisation du framework, avec des cas concrets que vous pouvez même utiliser comme base pour vos propres projets.

■ **Gauthier Delamarre**

Fiche technique

Nom de l'outil : Backbase
Éditeur : Backbase
Configuration : Windows XP / .NET 1.1
Site : www.backbase.com
Prix : Gratuit pour un usage personnel / à partir de 1 663 € pour un usage commercial
Configuration requise : Processeur 1 Ghz, 256 Mo de RAM sur le serveur
Indépendant de la plate-forme (web)

Les +
- Quantité ET qualité des éléments d'interface graphique
- Dialecte XML assez simple à prendre en main
- Plug-ins de colorisation et de complétion pour différents IDE (VisualStudio, Eclipse...)
- Documentation et exemples très complets
- Versions .Net et Java disponibles

Les -
- Pas d'intégration avec les langages de scripts comme PHP ou Python
- Syntaxe parfois un peu confuse

Internet Explorer 7 : le renouveau

La version 6 d'Internet Explorer commençait à dater et ne faisait plus le poids devant la concurrence. En sortant la version 7, Microsoft contre-attaque avec un produit plus ergonomique, mieux sécurisé et surtout plus conforme aux standards.

Les onglets ainsi que la visualisation des pages par vignettes facilitent grandement la recherche et la navigation lorsque plusieurs pages sont ouvertes.

Il ne faut pas parler de la nouvelle version Internet Explorer 7, mais plutôt des nouvelles versions. En effet, deux versions bêta sont actuellement disponibles : une pour Vista, une autre pour XP. La version finale qui sera disponible en principe à la rentrée, devrait également tourner sous Windows 2003 SP1 et Windows XP 64 bits.

IE 7 ne doit pas être considéré comme une simple remise à niveau du navigateur. Cette version apporte beaucoup de nouveautés, tirées d'une part de la concurrence, avec les produits comme Firefox, et d'autre part des nouveaux besoins des utilisateurs. Ces nouveautés ne sont pas toutes apparentes au niveau de l'interface, mais relèvent plutôt de la mécanique interne et par conséquent les développeurs qui construisent des interfaces Web, ou bien qui créent des extensions pour Internet Explorer devront en tenir compte. Pour Microsoft, ces changements s'adressent à 5 types de publics bien différents. En premier lieu, les utilisateurs bien sûr. C'est pour eux que les modifications sont le plus visibles, ensuite, il y a les concepteurs de sites Web, les développeurs de sites, les administrateurs réseau et enfin, les sociétés tierces parties qui développent des Add-ons. Ces changements peuvent être classés en trois grandes catégories. La première concerne la sécurité, avec en particulier un mode protégé dans lequel le navigateur est totalement isolé des autres applications et un système antiphishing. Vient ensuite l'ergonomie, avec la présence d'onglets pour chaque page ouverte et

(...)

Perforce.
Le système de GCL rapide.

Pour les développeurs qui n'aiment pas attendre.

[Rapide] [Évolutif] [Distribué]

Vous en avez assez d'utiliser un système de gestion de configuration logicielle qui vous empêche de soumettre vos modifications ? Le système de GCL Perforce est très différent : aussi rapide que puissant, aussi convivial qu'efficace. Perforce travaille à votre rythme.

Les performances de Perforce reposent sur trois piliers de conception : une base de données relationnelle indexée avec soin qui vous garantit un temps de réponse rapide pour les petites opérations et un haut débit lorsque les requêtes sont importantes, portant sur plusieurs millions de fichiers ; un protocole de transmission réseau continue des plus efficaces, qui minimise les temps de réponse et exploite au maximum les avantages de la bande passante ; enfin, un modèle de données intelligent centré serveur qui garantit l'incroyable vitesse de fonctionnement de la base de données et du réseau.

C'est à vous de voir. Voulez-vous travailler ou attendre ?

PERFORCE
SOFTWARE

Téléchargez sans conditions une copie gratuite de Perforce sur www.perforce.com. Vous bénéficiez d'une assistance technique gratuite pendant votre période d'évaluation.

Toutes les marques commerciales citées dans le présent document sont des marques commerciales, déposées ou non, de leurs dépositaires respectifs.

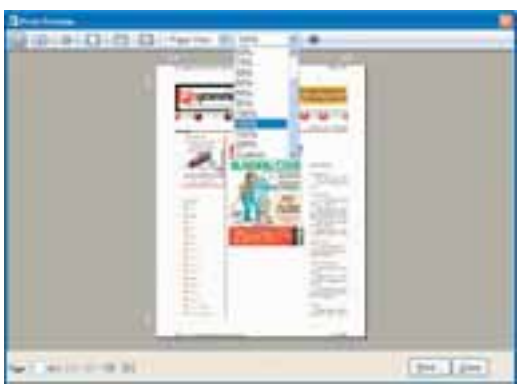
(...)

surtout une fonction de zoom sur la visualisation et l'impression. Dernier point enfin, qui intéressera plus particulièrement les concepteurs d'applications Web, c'est un meilleur respect des standards CSS 2 et AJAX.

Moins de bugs, mais plus de standards

Un des reproches que l'on peut faire aux précédentes versions est leur manque de compatibilité avec les feuilles de styles. Les versions 5 et 6 souffraient d'un problème avec les tags ‹div›, qui faisaient disparaître certains éléments. Ce bug disparaît. Une bonne douzaine d'autres bugs (répertoriés par exemple sur le site http://www.positioniseverything.net) ont également été corrigés. IE 7 est plus en conformité avec les standards CSS 2.1 et HTML 4.01 et supporte nativement XMLHTTPRequest ainsi que le canal alpha des images PNG qui permettent de définir le degré de transparence. Cette meilleure compatibilité syntaxique avec les autres navigateurs devrait permettre des configurations et des paramétrages de sécurité qui respectent les scénarios Ajax. Pour lutter contre les problèmes générés par les URL volontairement mal écrites c'est-à-dire avec des caractères spéciaux destinés à tromper les parsers, une nouvelle interface a été créée sous le nom de IURi (Uniform Resource Identifiers).

IE 7 est installé par défaut dans Vista ; pour Windows XP, il faut savoir que l'installation de cette bêta 2 ne peut se faire qu'avec une version anglaise du système et avec une licence valide, c'est-à-dire une copie dûment enregistrée. L'installation en profite pour demander l'installation de Microsoft Malicious Software Removal Tool. Ensuite, il faut mettre en marche le filtre de Phishing et la langue utilisée. Dans la rubrique France, il y a même le choix entre l'Alsacien, le Breton, le Corse ou l'Occitan... Dernier point, il faut désinstaller la bêta 1 lorsque vous installez la bêta 2.



Il est enfin possible, grâce à la fonction de Zoom d'imprimer correctement les pages Web.

Une barre d'onglet déjà vue quelque part

Au niveau de l'interface, les deux versions XP et Vista sont identiques. Ce que l'on note au premier abord c'est une ergonomie nettement améliorée, en particulier grâce à la présence de ces fameux onglets, qui permettent de passer instantanément d'une page à une autre, sans être obligé d'ouvrir plusieurs instances du navigateur comme c'est le cas actuellement. De plus, avec le bouton Quick Tab, toutes les pages ouvertes peuvent être visualisées en même temps sous forme de vignettes. Cela permet de sélectionner immédiatement et facilement celle que l'on recherche. Les barres d'outils sont bien moins encombrantes et grâce à OpenSearch 1.1 les fonctions de recherche sont désormais accessibles directement, avec en plus la possibilité d'y intégrer tous les moteurs de son choix. Ces recherches pouvant s'effectuer aussi bien sur les pages html que sur le RSS. Une autre nouveauté est l'accès direct sans passer par un logiciel externe, aux flux RSS. Un nouveau bouton nommé Feed Discovery permet d'en fournir la liste.

Ce qui nous a paru le plus intéressant est la fonction de Zoom qui permet de réduire les pages jusqu'à 30 % ou de les agrandir jusqu'à 999 % aussi bien en visualisation qu'en impression. De nombreux concepteurs ajustent leurs pages pour une résolution d'écran donnée et grâce à cette fonction, la visualisation pourra se faire dans les meilleures conditions. Le plus important c'est qu'il sera enfin possible d'imprimer les pages Web correctement, sans sauter les fins de lignes et sans gaspiller de feuilles inutilement en imprimant des pages à une seule ligne. Il est également possible à partir d'un simple clic sur la Barre d'outils, de choisir l'impression en mode portrait, ou en mode page ainsi que d'imprimer ou non les en-têtes et les bas de page.

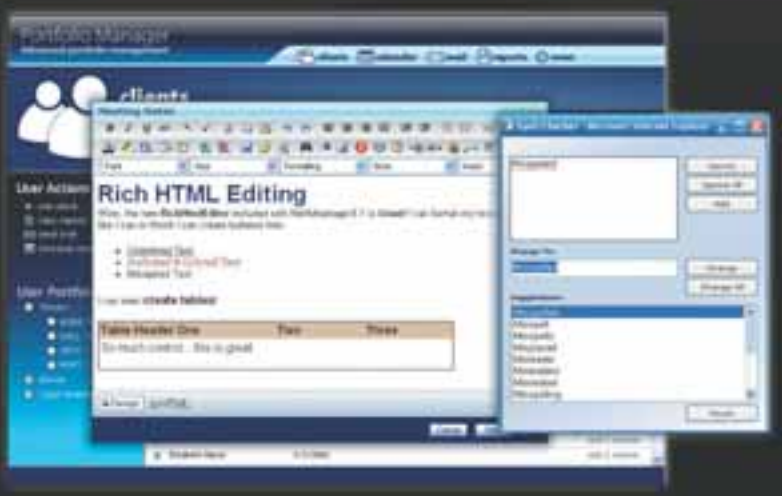
La fin des installations silencieuses

C'est surtout la sécurité qui a fait l'objet de toutes les attentions. IE 7 peut tourner dans un mode appelé mode protégé. Dans celui-ci, le navigateur est totalement isolé des autres applications et n'a la possibilité d'écrire que dans les fichiers temporaires du dossier Temporary Internet Files. Sinon, il devra obtenir le consentement explicite de l'utilisateur. Cette fonction doit empêcher les chargements et les exécutions intempestives des virus et leur rendre la Base de registres inaccessible. Ce mode protégé s'appuie sur le système de sécurité de Vista MIC (Mandatory Integrity Control) qui limite les opérations d'écriture et n'est donc pas disponible dans la version Windows XP. Comme avec le SP2 de Windows XP, il n'y a plus de possibilité d'installation silencieuse de code. Une autorisation d'exécution est systématiquement demandée à l'utilisateur. Les ActiveX ne peuvent être installés que sous le compte administrateur et ils sont désactivés par défaut, même s'ils sont déjà présents sur l'ordinateur. De plus, une autorisation est demandée lors de leur première exécution. Il existe donc plusieurs niveaux de sécurité. Un niveau Administrateur pour l'installation des ActiveX, un niveau utilisateur pour l'installation des Barres d'outils et des Add-ons et enfin, un bas niveau pour la configuration du navigateur et l'écriture des fichiers temporaires. Avec cette nouvelle version, l'utilisateur pourra donc contrôler les add-ons et aussi les certificats SSL de manière beaucoup plus simple qu'aujourd'hui. Ces améliorations devraient prévenir les installations de virus et de tous les codes qui exploitent les failles de sécurité, ou les « buffer overrun ».

Les nouveaux services antiphishing

Toujours au niveau de la sécurité, IE7 est doté d'un service antiphishing capable d'avertir l'utilisateur lorsqu'un site douteux apparaît. Son adresse apparaît alors en rouge dans la barre d'adresse. Pour cela, ce système se réfère à une base de données régulièrement mise à jour et qui regroupe tous ces sites. Pour limiter le nombre de faux positifs, l'utilisateur pourra également envoyer des messages pour signaler qu'un site est sain. Cette nouvelle version dispose aussi de contrôles parentaux qui, comme pour le mode protégé, s'appuient sur la sécurité de Vista et ne seront disponibles que dans cet environnement. Enfin, pour respecter la vie privée des utilisateurs, l'Historique et le cache peuvent être effacés en un seul clic. Les deux versions sont très proches les unes des autres, leurs interfaces sont identiques, seules changent les fonctions de sécurité propres sur l'architecture interne de Vista. Cette nouvelle version est séduisante, très ergonomique et devrait peut-être freiner un peu une concurrence qui commençait à se faire de plus en plus agressive.

■ **Alain KERHUEL**



2006 Vol. 1 Expanded Control Set

- Obtenez une expérience d'édition d'un HTML riche dans vos applications web en utilisant le nouvel éditeur HTML pour ASP.NET – fournissant à vos clients une interface éditeur facile à utiliser, similaire à Microsoft Word
- Ajoutez la fonctionnalité de correcteur d'orthographe dans vos applications web avec notre nouveau ASP.NET Spell Checker – ajoutez facilement le support du correcteur d'orthographe aux contrôles standards d'entrées de données ainsi qu'aux éditeurs d'Infragistics plus avancés
- Affichez et organisez facilement des données multi-champs dans vos applications Windows Forms avec le nouveau contrôle List View avec soutien complet du NetAdvantage PLF UIElement
- Insérez des hyperlinks dans vos applications Windows Forms en utilisant le contrôle Link Editor – utilisez le designer Link Editor afin d'éditer et de formater des hyperlinks en mode design

Expanded Control Offering

Nouveaux contrôles améliorant les caractéristiques et les types d'applications que vous pouvez livrer à vos clients.

NetAdvantage™ 2006 Vol. 1

L'ultime boite à outils pour créer rapidement l'interface utilisateur de vos applications Windows Forms, ASP.NET et Tablet PC.

NetAdvantage vous offre les outils vous permettant de vous exprimer avec:

- plus de 100 contrôles pour le développement d'interface utilisateur pour Win Forms et Web Forms
- designers intégrés dans Visual Studio et Smart Tags afin de programmer visuellement, composition des données et rendu de vos interfaces utilisateur
- Métaphores UI pour Office, XP, Vista, Microsoft CRM, Outlook, Portails, Docking, Dashboards et bien plus en appliquant simplement une de nos présélections en mode design
- Haute performance AJAX pour les contrôles WEB
- Construit sur un framework garantissant consistance et promouvant
  - La réutilisation du code
  - Temps réduit d'apprentissage pour le développeur
  - Maintenance facile de vos applications
  - Une expérience productive des utilisateurs de vos applications
- 3 nouvelles versions par an - vous informant des dernières technologies
- Code source fournit pour tous les contrôles et designers*
- Support mondial grâce à nos équipes de support technique aux développeurs sur trois continents (Amérique du Nord, Europe et Asie)

TestAdvantage™

Le premier outil prêt-à-utiliser automatisant le test des interfaces utilisateur en supplément du QuickTest Professional de Mercury afin de supporter les interfaces utilisateur Windows Forms développées avec NetAdvantage.

www.infragistics.com/testadvantage

Training

Apprenez comment implanter efficacement la technologie Infragistics pour interfaces utilisateur selon votre domaine d'expertise pour le développement de vos applications.

www.infragistics.com/training

Essayez dès aujourd'hui NetAdvantage et donnez facilement à vos applications une interface utilisateur riche et irrésistible.

www.infragistics.com/download

NetAdvantage avec Souscription et Support Prioritaire (875 €)* NetAdvantage avec Souscription (699€)*
*la souscription inclut tous les éléments ainsi que toutes les mises à jour, code source C# complet pour tous les éléments .NET et designers pendant un an, renouvelable.
Prix hors taxes et susceptibles d'être modifiés.

N° Vert 0800 667 307 • sales-europe@infragistics.com

Feriscope
L'intranet de RFF, revu et refondu

RFF (Réseau Ferré de France) a fait évoluer son Intranet pour en faire un portail d'information, de communication et de travail collaboratif orienté utilisateur. Un défi intéressant pour SQLI qui, en plus, a conquis sa certification qualité CMM-I de niveau 3 !

"Nous tenions à faire du portail Feriscope une plate-forme fédératrice d'information, de communication et de travail collaboratif, dont chacun des collaborateurs, quel que soit son profil, pourrait tirer parti au quotidien", indique Isabelle Million, responsable de la communication interne de RFF, chef de projet maîtrise d'ouvrage. Plusieurs arguments ont convaincu RFF de choisir SQLI, notamment l'expertise acquise aux cotés d'Airbus, depuis 2003, sur la mise en œuvre de solutions Livelink.

Feriscope, l'intranet de RFF.

Un travail en partenariat

« Nous avions un comité de projet une fois par semaine et un comité de pilotage une fois par mois qui nous permettaient de suivre exactement l'évolution du travail. Je les ai pressés, les rapports étaient excellents et le travail fini en temps et en heure », raconte Isabelle Million.

Avec quatre objectifs majeurs : offrir aux collaborateurs de RFF un accès rapide et intuitif à l'information et mettre à leur disposition des outils de travail collaboratif tels que l'annuaire, des agendas partagés ou un forum. En second lieu, permettre aux administrateurs de créer des groupes thématiques, tout en assurant la gestion des droits d'accès et la fiabilité de l'information via un système de workflow de validation des publications. Troisième point : mettre à la disposition des contributeurs un outil de gestion de contenu leur permettant de publier en toute simplicité de l'information, mais aussi de stocker et de partager des informations. Enfin, fédérer l'ensemble des collaborateurs de RFF autour d'un portail se posant à la fois en point d'accès unique à l'information et aux différentes applications métier et bureautiques.

Pour ce faire, un programme marathon : début des développements en février 2005, trois semaines de recette en juin et le passage en production, début juillet.

Autres conditions non négociables du cahier des charges : utiliser la solution Livelink Communities of Practice d'OpenText, choisie en amont par RFF. Conséquence : un travail en partenariat, cette fois-ci avec les équipes d'OpenText dans le cadre d'une problématique de portail et de GED – réalisé par l'éditeur – avec comme objectif, une interopérabilité totale.

OpenText, éditeur de la solution Livelink

C'est l'un des acteurs majeurs dans le domaine de l'ECM (Enterprise Content Management). Livelink a été développé dans le langage propriétaire Oscript et il est construit de façon complètement modulaire.

Une intéressante application des « communautés »

Le projet est officiellement lancé le 1er décembre. *« L'un des défis était l'utilisation de la technologie éditeur alors que les ressources n'existaient pas »*, souligne Jérôme Lavancier, chef de projet SQLI, entouré d'un directeur de projet et de deux ingénieurs.

Sitôt l'analyse fonctionnelle du périmètre faite, le studio graphique, sqli agency, prend le relais. Une piste graphique est retenue par RFF, la maquette suit.

L'équipe entre alors dans la période de développement. Elle dure 3 mois et inclut des tests unitaires permettant de valider l'ensemble des modules additionnels constituant le nouveau portail Feriscope. *« Dans le cadre de l'intégration pour RFF, le module additionnel des communautés de pratique ajoute une dimension collaborative au socle de base, plutôt orienté GED. Un autre module a également été déployé : Directory Services qui permet de s'intégrer de façon transparente avec l'annuaire d'entreprise Active Directory de RFF »*, spécifie J. Lavancier.

J. Lavancier, chef de projet SQLI

Mais le travail avec la solution imposée était-il évident ? *« OpenText propose aux intégrateurs un SDK dédié permettant de développer des modules additionnels complètement intégrés avec l'environnement Livelink, répond*

Jérôme Lavancier. D'ailleurs, Feriscope s'appuie sur un ensemble de modules additionnels, intégrant notamment l'éditeur HTML WYSIWYG FCKEditor pour la gestion du contenu Web. Un autre module prend en charge toute la gestion et les interactions de la partie annuaire du portail, dont les données sont gérées dans un autre système d'information chez RFF ».

Le schéma d'architecture original. La seule différence dans la réalisation est qu'il y a deux serveurs web Livelink frontaux et non un.

Au final, la recette, est confiée à un prestataire qui valide le portail. Et dans le même temps, le projet est audité dans le cadre de la certification CMM-I de niveau 2 garantissant à RFF un haut niveau de qualité sur cette réalisation. Après ce double « Go », le lancement est retardé... pour cause de grandes vacances, qui permettent toutefois à RFF de finaliser le contenu de son nouveau portail lancé le 1er décembre 2005. Mission accomplie : aujourd'hui, une quarantaine de contributeurs alimentent quotidiennement Feriscope.

RFF : la gestion du réseau ferré national

Créé en 1997, RFF est propriétaire et gestionnaire du réseau ferroviaire français, jusque là géré par la SNCF, en contrepartie de la dette ferroviaire qu'il a reprise. Il a pour objet l'aménagement, le développement, la cohérence et la mise en valeur du réseau ferré national. C'est un Épic (établissement public à caractère industriel et commercial, comme la SNCF) : il ne peut donc ni être mis en faillite, ni faire l'objet d'une saisie.Le président Michel Boyon est entouré de 700 collaborateurs.

Un réseau de 12 directions régionales a été mis en place dans le but de mieux répondre aux besoins des collectivités territoriales, notamment les régions. Elles interviennent directement dans la gestion du réseau (utilisation, exploitation et entretien), dans sa modernisation et son développement (volets ferroviaires des contrats de plan État-régions, projets de lignes ou de liaisons nouvelles, suppression des passages à niveaux, bruits, pôles d'échanges), ainsi que sur les opérations domaniales et de valorisation du patrimoine RFF.

■ Claire Padych

Nat System présente
la 1re solution
AGL J2EE
Web 2.0

NatIxt

"des applications web dynamiques, communicantes, riches"

"une ergonomie plus souple, plus interactive"...

Avec NatIxt, vos pages web ne se contentent plus d'afficher des données : elles deviennent des conteneurs dans lesquels vous pouvez mettre à jour et différencier les contenus, les fonctions, selon les zones.

A vous les progrès en productivité, en simplicité.

Une démo ? Rendez-vous sur notre site www.natsystem.fr !

Nat System
Stratégiquement vôtre

Nat System - Immeuble Le Péricentre - 16 av. Louison Bobet - 94120 Fontenay-sous-Bois - www.natsystem.fr

Ressources complémentaires
Programmez.com
sources, liens, téléchargement, forums

Peer to Peer

Le réseau nouvelle génération

1re partie

Depuis longtemps, les medias de masse traitent du P2P (Peer to Peer) comme d'un quasi fléau. Les majors de musique ou de films s'acharnent dessus. Or, cette utilisation illégale du P2P, ou plutôt, des clients P2P, masque le réel intérêt de cette technologie.

Le P2P est, il ne faut pas l'oublier, un formidable outil de communication, comme l'est le téléphone, le mail, le fax, etc. Surtout, il propose un modèle d'architecture totalement différent du classique client / serveur et du fameux serveur central. Le P2P sert à beaucoup de choses : partager, échanger et accéder à des fichiers distants, messagerie instantanée, créer des réseaux collaboratifs, stocker des données, faire de la messagerie instantanée, voire des grilles de calculs, ou servir de service de déploiement d'applications.

Technologie mature, usage multiple !

Certains présentent le P2P comme l'avenir d'Internet. Nous dirons, pour notre part, que d'une certaine manière, il fait partie de la mouvance Web 2.0, notamment pour l'esprit collaboratif. Surtout, le P2P offre des perspectives gigantesques d'interconnexion entre terminaux, voire entre applications, via une simple interface cliente. Ce n'est pas un hasard si des éditeurs comme Sun ou Microsoft travaillent sur le sujet activement et s'intègrent au cœur des systèmes comme Windows. Le P2P intéresse tout le monde, de l'utilisateur à la grande entreprise.

Des sociétés se sont créées sur ce marché, telle Aelitis, spécialisée dans le P2P, éditeur du logiciel -largement downloadé- Azureus, fonctionnant sur le protocole BitTorrent. La société est également prestataire et intégrateur (www.aelitis.com).

Dans ce dossier en deux parties, nous allons essayer de vous montrer le vrai visage du P2P, dans les scénarios d'utilisations, les projets des éditeurs, et surtout dans les diversités des protocoles P2P du marché.

■ François Tonic

Attention à la loi : halte à l'échange de fichiers piratés !

A l'heure où nous bouclons la revue, le sénat doit examiner la loi, votée le 21 mars à l'Assemblée, destinée à lutter contre le piratage numérique, en particulier musiques et films. En l'état actuel, le téléchargement illégal est puni d'une amende de 38 €, portée à 150 € si les fichiers sont mis à disposition du public.

Mais surtout, les développeurs de « logiciels conçus pour l'échange de fichiers non autorisés » risquent 300 000 € d'amende et 3 ans de prison ! Mais peut-on sanctionner l'outil (le logiciel) alors que c'est l'usage de certains (le piratage) qui est répréhensible ? Le rapporteur de la loi, Christian Vanneste, précise en effet : « il s'agit de favoriser le téléchargement licite respectant le droit des auteurs »

Sécurité des réseaux P2P : fantasme ou réalité ?

FORUM P2P
www.programmez.com

L'insécurité informatique est une réalité du quotidien. La question peut aussi se poser pour les réseaux P2P et leur utilisation. Mais finalement, peut être se trompe-t-on de problème !

Dans un réseau P2P, il faut différencier le protocole, l'application, les fichiers et le contenu. La technologie P2P n'est pas en soi en cause, comme le rappelle Gilles Gravier de Sun (chief Technology Strategist for Security). Le véritable problème vient du contenu, des fichiers que l'on échange. Si l'utilisateur accepte des fichiers non certifiés, non signés, non vérifiés d'un utilisateur inconnu, il le fait à ses risques et périls, car il est très simple d'injecter, durant la transmission, des blocs corrompus ou malicieux. Le fichier peut dès le départ contenir un malware. En usage professionnel ou même entre amis, il convient donc de respecter certaines règles élémentaires de sécurité, que les bons protocoles P2P proposent par défaut :

- ne jamais accepter un fichier de provenance inconnue, sans au moins l'avoir vérifié,
- utiliser de préférence des fichiers vérifiées et certifiés (checksum par exemple),
- vérifier l'identité de l'utilisateur envoyant le fichier.

Il faut pouvoir assurer les piliers de base :

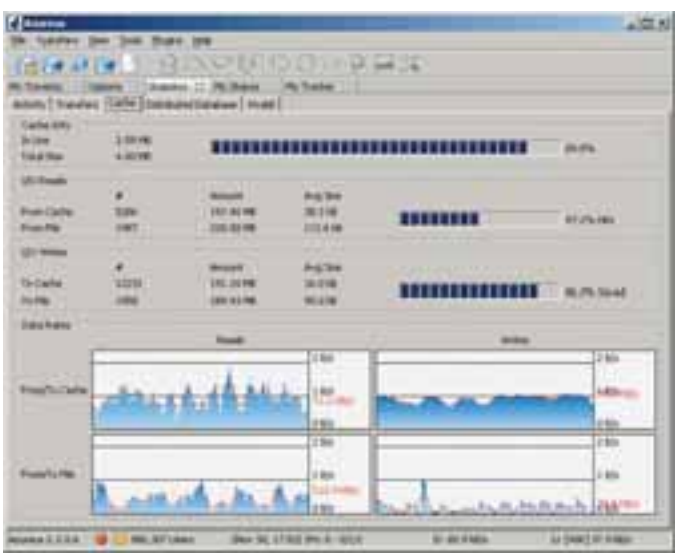
- confidentialité : les utilisateurs non autorisés ne doivent pas accéder aux données « protégées »,
- autorisation : garantir que l'envoyeur est bien autorisé à envoyer, s'assurer que tout envoi est fait par un utilisateur proprement identifié et que le receveur le soit aussi,
- intégrité des données.

Cependant, le P2P n'étant pas exempt de failles, de bugs, il est toujours possible qu'un hacker puisse exploiter une faiblesse d'un client P2P pour s'infiltrer ou envoyer un code malicieux. Dans ce cas, surveillez de près les alertes et les patches de sécurité.

Si vous prenez, JXTA, dès le départ, la position du projet était d'être le plus neutre possible sur la cryptographie et la sécurité. Au développeur d'utiliser les mécanismes qu'il souhaite.

Des solutions limitées ?

Pour garantir (un minimum) le fichier (sa taille, son intégrité), passez par un classique checksum MD5, les principaux protocoles P2P l'implémentent. « Les risques de collision [dans le cadre d'un checksum MD5] sont rares, mais cela commence à apparaître » précise Gilles Gravier. Donc, il s'agit d'une approche de base que l'on doit coupler à d'autres mécanismes : signature, cryptage de bout en bout. Pour l'utilisateur, notamment en entreprise dans le cadre d'un P2P interne / externe, il sera intéressant de coupler le client P2P à un annuaire pour authentifier l'utilisateur. Skype utilise ce principe. Mais finalement, il n'y a pas grand-chose à faire de plus, les mécanismes demeurent très classiques. Optez de préférence pour du P2P reconnu par le marché, évitez les clients exotiques.



La sécurité du P2P peut aussi découler du choix que l'on fait entre un format ouvert et un format propriétaire. Nous ne disons pas que l'un est mieux paré que l'autre mais question réactivité et correctifs, il peut y avoir une différence. De plus, si vous utilisez du P2P entre différents types de terminaux, les problématiques de sécurité peuvent changer. Il faudra prévoir des authentifications d'accès, de connexions, par exemple pour le P2P mobile (idem en utilisation wifi) et éviter de laisser en clair toute information liée à son client P2P. Classiquement, vous pourrez opter pour du VPN pour créer un tunnel de session sécurisé de bout en bout. Pour l'identification, une des méthodes sera de passer par exemple par l'adresse IP du PC utilisé ou le numéro SIM pour un portable. Mais, s'il y a possibilité de connecter de n'importe où, il faudra utiliser une méthode d'authentification différente. Le PKI est possible dans bien des solutions P2P comme JXTA. Il existe des logiciels permettant de crypter les données transitant dans un P2P, ex. : Steganos P2P.

Bien entendu, on dira que le P2P passe à travers le firewall, mais comme le dit M. Gravier, « et alors ? ». D'autres technologies passent aussi à travers. Le P2P est difficile à bloquer en soi. Skype, qui fait partie de ce type de technologie, illustre ce « problème », car avec Skype, il est très difficile d'identifier les paquets en entrée / sortie.

François Tonic

FORUM P2P
www.programmez.com

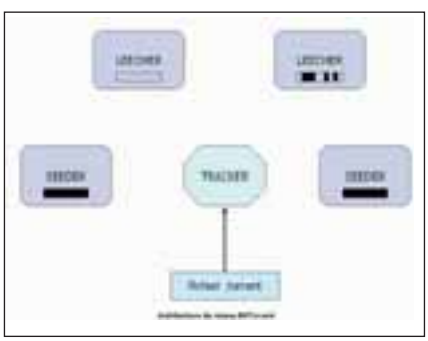
BitTorrent au service de l'industrie et des professionnels !

Les entreprises commencent de plus en plus à remplacer le téléchargement FTP par celui, plus efficace, proposé par BitTorrent.

1 Le principe et le fonctionnement de base pratique

A l'heure où nous rédigeons ces lignes, SuSE vient d'annoncer la sortie de la bêta 8 de SuSE 10.1. Ce genre d'annonce provoque toujours, même s'il s'agit d'une version non stable, un afflux de demandes de téléchargements. En un court laps de temps, de très nombreuses personnes vont vouloir télécharger les images ISO de cette distribution. Il y a quelques années, l'industrie du logiciel n'avait qu'une seule réponse : proposer un serveur FTP suffisamment puissant pour résister à la demande (avec une série de serveurs FTP miroirs, mais le problème posé reste le même : comment résister à la charge ?). Depuis peu, d'autres techniques ont vu le jour. D'abord, il est possible de ne rapatrier que les différences (delta iso) entre l'image ISO d'une version antérieure que possède déjà l'utilisateur et la nouvelle version. Ensuite, et surtout, il est possible de mettre à disposition de tous des fichiers torrent (http://download.opensuse.org/distribution/SL-10.1-beta8/SUSE-Linux-10.1-beta6_beta8-i386.torrent). Bittorrent est un protocole P2P créé par Bram Cohen en automne 2002.

Celui-ci (le .torrent) contient des métadonnées de l'image ISO à télécharger telles que le nom du fichier, sa taille, la somme de contrôle de chaque unité, et l'adresse du tracker. Nous allons revenir sur ces termes, mais ce .torrent a été généré par SuSE avec un utilitaire comme btmakemetafile. Le fichier ISO à partager a été tronçonné en unités moins encombrantes pesant quelques centaines de Ko. Comme avantage, le serveur d'origine (appelé "seed" ou graine) n'aura à partager chaque unité qu'une seule fois ! En effet, chaque client torrent qui rapatriera une unité pourra à son tour partager celle-ci avec d'autres. Au niveau de la charge et de la bande passante, l'industrie du logiciel a donc tout intérêt à adopter le chargement par BitTorrent, car ce protocole est beaucoup moins exigeant en terme de ressources. C'est moins cher et ce mécanisme P2P est capable de résister très fortement à la demande sur un temps très court (en quelque sorte, le goulot d'étranglement des performances se déplace de l'entreprise vers l'Internet). En clair, chaque client devient à son tour un seed pour un autre (pair), et le serveur d'origine (celui de SuSE) voit ses requêtes baisser très rapidement. Au point de vue technique il est nécessaire de se reposer sur un serveur central, nommé tracker, pour gérer la coordination entre les seeders et les clients (aussi appelés "leechers").



Le fichier torrent (un fichier binaire de quelques ko) associé au fichier que vous essayez d'obtenir sera donc envoyé sur un tracker, et le seeder « premier du nom » commencera à rendre les unités disponibles au réseau P2P. En réalité, la machinerie logicielle est un peu plus complexe : par exemple, les clients récupéreront en priorité les unités les plus rares, afin d'augmenter leur visibilité.

BitTorrent a été écrit en Python, ce qui permet de l'utiliser sur de nombreuse plates-formes, dont Linux et Windows. Concrètement, le serveur génère un fichier .torrent que les clients vont télécharger (le client vous demande alors où vous désirez sauvegarder le fichier).

Signalons que la société Prodigem (http://www.prodigem.com/) propose un système basé sur BitTorrent pour diffuser des données à un coût réduit. En effet, si vous payez 1 dollar, vous bénéficiez d'un espace de stockage de 100 Mo et de 3 Go de bande passante, mais si 3 utilisateurs à leur tour relaient les données, la bande passante de Prodigem ne sera plus mise à contribution (pour le prix de 14,95 $ par mois, vous disposez d'un espace de stockage de 1,5 Go et jusqu'à 45 Go de bande passante).

Pour créer un tracker
bttrack.py --port 6969 --dfile dstate
dstate est un fichier qui conservera les informations de votre serveur entre chaque lancement. Si vous pointez votre navigateur à l'adresse http://localhost:6969, vous obtiendrez l'état actuel du tracker.
Pour générer un .torrent :
btmakemetafile.py LeFichierAPartager
http://localhost:6969/announce
Vous pouvez évidemment remplacer l'adresse localhost par celle fixe de votre tracker. Pour que d'autres puissent accéder au fichier et le télécharger, il ne vous reste plus qu'à mettre à disposition de tous (ou bien de certains) le .torrent ainsi généré.

2 Les utilitaires et les API

L'implémentation canonique du protocole a été rédigée en Python (lib BItTorrent se trouvant dans le répertoire BitTorrent), certains planchent sur une implémentation en C++ et en C# mais pour l'instant, sans résultat

SUIVEZ LA VOIE DE L'AUTOMATISATION

Nouvelle version
LEONARDI V3.2
avec
support AJAX transparent
automatisation des tests

Et programmez moins!
Réduisez jusqu'à 70% les délais
et coûts de réalisation de vos applications

Leonardi

Logiciel Client Universel

Téléchargez la nouvelle version gratuite
multi-afficheurs
sur notre site www.lyria.com

Leonardi est un framework JAVA/XML.
Son principe de fonctionnement "model driven"
permet de créer des applications prêtes à l'emploi
à partir d'un simple modèle de données.

- Crée les IHM à la volée en fonction de la technologie d'affichage choisie : HTML, DHTML, SWT, AWT, SWING, SVG.
- Support transparent d'AJAX, intégration plate-forme Eclipse, déploiement via JavaWebStart...
- Connexion aux principales bases de données du marché : Oracle, DB2, MS SQL Server, Sybase, MySql, PostgreSQL...
- S'intègre dans votre système d'information grâce à des modules de connexion pour JMS, LDAP et serveurs d'applications.
- Gestion de la persistance des données et des transactions.
- Solution portable sur toute plate-forme (Windows, Linux, Unix, Mac OS...) grâce à l'utilisation du langage JAVA.
- Livré avec Leonardi Studio, environnement de développement intégré dans Eclipse.
- Implémentation du code métier en JAVA.
- Import de modèles de données UML à l'aide de XMI.
- Différents types de vues disponibles :
  - formulaires de consultation et modification,
  - tableaux,
  - vues arborescentes,
  - diagrammes de Gantt,
  - vues synoptiques,
  - vues statistiques...

Déployez votre application en WEB,
client lourd ou sous forme de plugin Eclipse
sans aucune modification!

Client Software Solutions

www.lyria.com

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.

Leonardi est une marque de LYRIA S.A. Les autres marques citées dans ce document sont des marques déposées par leurs propriétaires respectifs.

probant (http://sourceforge.net/projects/btpp/ pour le C++, et http://sourceforge.net/projects/csharpbt/ pour le C#). Des modules PERL ont été mis au point (http://bklyn.org/~cae/Bencode.pm et http://www.gluelogic.com/code/BitTorrent/TrackerCGI.pm) ainsi qu'un sniffer qui permet de connaître l'état d'un torrent (http://www.highprogrammer.com/alan/perl/torrentsniff.html):

./TorrentSniff.plhttp://www.rpmfind.net/BitTorrent/shrike-i386-disc3.iso.torrent

```
http://www.rpmfind.net/BitTorrent/
shrike-i386-disc3.iso.torrent
   info hash:      b8d1e315139bdc
5c024285aa65efc1170f7535a3
   announce url:   http://www.rpmfind.net:
6969/announce
   full copies:    4 seeds
   partial copies: 5 leeches
            Bytes File
      508,592,128 shrike-i386-disc3.iso
```

Il existe aussi une API en Ruby disponible à l'adresse http://rubytorrent.rubyforge.org/.

Voici d'ailleurs un très court exemple d'un client BitTorrent en Ruby :

```
require 'rubytorrent'
bt = RubyTorrent::BitTorrent.new(ARGV.shift)
thread = Thread.new do
  while true
    puts bt.percent_completed
    sleep 15
  end
end
bt.on_event(self, :complete) { thread.kill }
thread.join
```

Un programmeur a aussi développé en Ruby un générateur de torrent. Cet outil se nomme Aversa et est disponible à l'adresse http://aversa.rubyforge.org/./aversa.rb --makemetafile wxruby.exe,http://rubyforge.org:6969/announce

```
announce http://rubyforge.org:6969/
announce
created by
Aversa (http://aversa.rubyforge.org) v0.3
creation date Mon Aug 30 10:44:29
EDT 2004
info
 length 1424864
 name wxruby.exe
 piece length 262144
 pieces [6 SHA checksums]
```

BitTorrent utilise le protocole HTTP pour récupérer les méta-données et la liste des peers, tandis que la communication entre les pairs s'établit via un protocole binaire dédié. C'est le protocole bencode, et bdecode qui est utilisé. Cet encodage est également employé au niveau de la réponse du Tracker, suite à une requête d'un client. Ce format fonctionne par type de valeurs (Integer,String, List et Dictionary). Vous trouverez ci-dessous un exemple simple nommé detrackertorrent qui a été programmé en Python et qui emploie ce protocole de codage/décodage :

```
#!/usr/bin/python
# The contents of this file are subject to
the BitTorrent Open Source License
# Version 1.1 (the License).  You may not
copy or use this file, in either
# source code or executable form,
except in compliance with the License.  You
# may obtain a copy of the License
at http://www.bittorrent.com/license/.
#
# Software distributed under the License is
distributed on an AS IS basis,
# WITHOUT WARRANTY OF ANY
KIND, either express or implied.
See the License
# for the specific language governing
rights and limitations under the
# License.
# Written by Olivier Crête
# Based on changetracker-console
by Henry 'Pi' James and Bram Cohen

from BitTorrent.platform import install_translation
install_translation()

from os.path import basename
from sys import argv, exit
from BitTorrent.bencode import bencode,
bdecode

if len(argv) < 3:
    print "Usage: %s TORRENTFILE
client:port [client:port ...]
" % basename(argv[0])
    print
    exit(2) # common exit code for syntax error

f= argv[1];
h = open(f, 'rb')
metainfo = bdecode(h.read())
h.close()

if metainfo['announce']:
    print _("old announce for %s: %s")
% (f, metainfo['announce'])
    del metainfo['announce'];
    metainfo['nodes'] =  [(a[0], int(a[1]))
for a in [node.strip().split(":") for node
in argv[2:]]]
    h = open(f, 'wb')
    h.write(bencode(metainfo))
    h.close()
else:
    print "The torrent was already trackerless"
```

3 Conclusion et avenir

Comme les API BitTorrent dans un cadre .net sont seulement en préparation, le plus simple pour l'instant pour un programmeur est de travailler en Python. Ce qui n'est pas un problème, vu que nous vous rappelons qu'un module Python peut facilement être appelé depuis un autre langage comme le C. BitTorrent évolue aussi. D'une part, actuellement, l'utilisation de fichiers RSS permet de diffuser très rapidement des fichiers .torrent dès qu'ils sont disponibles sur le réseau. D'autre part, Bram Cohen planche sur la prochaine génération du réseau BitTorrent. Ce BitTorrent 2 (officiellement «bt2») supportera le protocole UDP au niveau du Tracker, et interconnectera l'ensemble des serveurs en vue de redistribuer dynamiquement la charge.

Avantages :

- le protocole le plus populaire actuellement mais non dans l'industrie (ou on lui préfèrera de loin JXTA) ;
- très rapide, protocole libre dont le code source est disponible en Python ;

Inconvénients :

- Pas d'API.net ;
- Si au début de la disponibilité du fichier, de nombreuses sources sont disponibles (car le fichier est populaire) ce ne sera plus le cas au bout d'un temps relativement court (quelques jours ou semaines) car il devient de moins en moins partagé (notez que la solution Prodigem y remédie).

■ Xavier Leclercq
Xavier.Leclercq@programmez.com

Le framework P2P JXTA

Le projet JXTA a pour objectif de servir de couche de base pour la définition de services pair-à-pair spécialisés. Avec JXTA, le développeur peut s'initier au P2P professionnel grâce à ce framework Peer-to-Peer Open Source.

Pratique

C'est en 2001 que Sun lança un projet Open Source visant l'interconnexion de n'importe quel système au-dessus de n'importe quel réseau, P2P compris. Aujourd'hui, 5 ans après son lancement, JXTA fédère plus de 25 000 membres actifs, dont de très nombreux développeurs travaillant sur une centaine de projets ! JXTA signifie Juxtaposition entre l'ancien modèle client/serveur et le N2N (noeud à noeud). Il s'agit vraisemblablement de la norme industrielle P2P la plus élaborée à l'heure actuelle (à la base d'applications Peer-to-Peer professionnelles). Une implémentation JXTA existe en Java avec un échange de données en XML ; ou plus exactement, JXTA offre des communications via des messages XML, permettant ainsi à n'importe quelle entité sur le réseau (téléphone cellulaire, serveur, PDA, etc.) de communiquer (*).

Un exemple concret d'utilisation de JXTA en P2P est le logiciel BitVault Network de la société 312, Inc. Ce software de sauvegarde permet d'envoyer sur un réseau P2P des fichiers cryptés. Les données sont stockées sur chacun des clients, mais ces derniers ne peuvent y accéder car ils sont protégés par chiffrement.

1 Comment travailler avec le shell JXTA

JXTA propose un ensemble de protocoles de base (six protocoles en tout) pour la découverte ou/et le monitoring des ressources, et la communication P2P. Il permet même au développeur de définir ses propres services. L'implémentation java de JXTA nécessite J2SE (Java 2 Standard Edition), ainsi que le J2RE (Java 2 runtime environment), et le SDK (Java 2 Software Developer's Kit). En pratique, et en admettant que vous ayez déjà un environnement java de développement complet et fonctionnel à votre disposition, vous devez commencer par récupérer les fichiers JXTA (si vous êtes sous Windows XP, surfez sur le site de JXTA et cliquez sur le lien "launch now" sous l'intitulé «Shell - (demonstration for software developers)»).

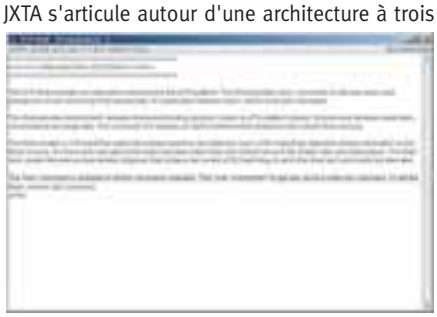
Après l'installation, on vous demandera de configurer l'application shell JXTA (cette opération de configuration initiale sera effectuée au moins une fois, quelle que soit l'application JXTA lancée). Indiquez simplement un nom pour votre réseau (ici «programmez») et un mot de passe comportant au moins 8 caractères. A chaque démarrage du shell, on vous demandera de vous identifier.

Bon à savoir : par défaut, le port JXTA est le 9701 et seul le protocole TCP/Multicast est sélectionné. Plus tard, vous pourrez aussi réaliser des essais en sélectionnant le protocole

de communication HTTP. Un fichier de configuration est généré automatiquement à la première exécution sous l'arborescence de l'application lancée (répertoire ".jxta/"). Si vous désirez reconfigurer l'application, il suffit d'effacer ce répertoire, puis de relancer l'application.

Le Shell JXTA est une application interactive qui vous autorise à lister l'environnement JXTA, et à partir duquel vous pouvez aussi taper des commandes "peer" qui permettent de commander et de contrôler les pairs du réseau.

JXTA s'articule autour d'une architecture à trois



couches : le noyau (JXTA Core), les services et les applications.

En fait, JXTA Core se décompose lui-même en quatre composantes : *Peer pipes* (capacité de connecter un pair à un autre et de partager les informations sur le réseau) ; *Peer groups* (la création de groupes de façon dynamique) ; *Peer monitoring* (surveillance, mesure de l'interaction et définition des politiques de contrôle entre pairs) et *Peer Sécurité* (garantie de la confidentialité, de l'identité et de l'accès). Le JXTA Shell est un outil middleware qui se situe entre les couches services et applications. Cet outil fonctionne à la manière d'un shell Unix : vous obtenez un «prompt» nommé «JXTA›» à partir duquel vous tapez des commandes. Soit la commande entrée produira une sortie écran, soit elle produira un objet en sortie. Chaque référence d'objet peut ainsi être stockée dans une variable d'environnement :

```
JXTA>env
History = "History"
[net.jxta.impl.shell.bin.history.HistoryQueue]
rootgroup =
"Default Group" [net.jxta.impl.peergroup.
RefCountPeerGroupInterface]
stdgroup =
"Default Group" [net.jxta.impl.peergroup.
RefCountPeerGroupInterface]
console = "console" [net.jxta.impl.shell.
SwingShellConsole]
consout = "Default Console
OutputPipe" [net.jxta.impl.shell.
ShellOutputPipe]
worldgroup =
"World Peer Group"
```

(*) En réalité les messages sont, soit présentés en XML pur, soit encodés en binaire en Base 64 dans le corps d'un message XML.

Comprendre la terminologie pour maîtriser JXTA

• Peer/Pair : noeud d'un réseau P2P qui peut aussi bien être une machine qu'un utilisateur.
• Peer group : un ensemble de peers offrant un service.
• Rendezvous peer : informations servant à découvrir d'autres peers ou ressources.
• Router peer : chemin de communication pour peers.
• Pipe : canal de communication asynchrone pour envoyer et recevoir des messages.
• Endpoint : destination pour laquelle un pipe a été lié.
• Advertisements : document en XML qui décrit un peer, un peer group, un pipe ou un service. Cela devient un peu compliqué, mais sachez qu'il existe 9 types d'advertisements : PeerAdvertisement, PeerGroupAdvertisement, Pipe Advertisement, Module Class Advertisement, Module Spec Advertisement, Module Impl Advertisement, Content Advertisement et PeerInfo Advertisement
• Service : fonctionnalité atteignable par un autre peer.
• Message : un message est envoyé à un pair par un pipe, sa structure est en XML et peut contenir des données (pipe.createOutputPipe (pipeAdv, this).

Voici le code de génération aléatoire d'un pipe de type «Advertisement» :

```
PipeAdvertisement myPipeAdv = null;
  try {
      myPipeAdv = (PipeAdvertisement)
    AdvertisementFactory.newAdvertisement
(PipeAdvertisement.getAdvertisementType());
    myPipeAdv.setPipeID(IDFactory.newPipeID
(netPeerGroup.getPeerGroupID()));

myPipeAdv.setName("Exemple_De_Creation_
PipeAdvertisement ");
      myPipeAdv.setType("JxtaUnicast");
  } catch (Exception e) {
      e.printStackTrace();
  }
  if (myPipeAdv == null) {
      System.out.println("waitptext:
ne peux créer un Pipe Advertisement");
  }
  System.out.println("creation d'un pipe
advertisement...");
  System.out.println(myPipeAdv);
```

```
[net.jxta.impl.peergroup.
PeerGroupInterface]
stdout =
"Default OutputPipe" [net.jxta.impl.shell.
ShellOutputPipe]
stdin =
"Default
InputPipe" [net.jxta.impl.shell.ShellInputPipe]
shell =
"Shell 1" [net.jxta.impl.shell.bin.Shell.Shell]
consin =
"Default Console InputPipe" [net.jxta.impl.
shell.ShellInputPipe]
```

Vous pouvez utiliser la commande "man" dans le shell JXTA pour obtenir plus de détails sur une commande. Voici quelques commandes :
• whoami (pour déterminer qui vous êtes) ;
• whoami -g (pour déterminer le nom du groupe courant) ;
• groups (liste des groupes connus) ;
• groups -r (recherche dynamique de groupes) ;
• join (liste des groupes joints) ;
• join -d legroupeajoindre (joindre un groupe) ;
• peers (listage des peers connus) ;
• peers -r (découverte dynamique de pairs) ;
• peers -p peerid (découverte d'un pair particulier connaissant son id) ;
• peers -a nom (découverte d'un pair particulier connaissant son nom) ;
• peers -f (efface le cache des pairs découverts) ;
• talk -register monnom puis talk
• login monnom et talk -u monom
• destinataire (entre en communication avec «destinataire»).

2 Compilation et exécution d'un exemple simple

L'exemple suivant est sans doute le plus simple figurant dans le kit de développement. Il instancie le «démon» JXTA et affiche le nom du groupe peer et son identifiant.

SimpleJxtaApp.java

```
import net.jxta.peergroup.PeerGroup;
import net.jxta.peergroup.PeerGroupFactory;
import net.jxta.exception.PeerGroupException;

public class SimpleJxtaApp {

    static PeerGroup netPeerGroup = null;

    public static void main(String args[]) {

        System.out.println("Starting JXTA ....");
        SimpleJxtaApp myapp =
new SimpleJxtaApp();
        myapp.startJxta();

        System.out.println("Hello from JXTA
group " +
                netPeerGroup.getPeerGroup
Name() );
        System.out.println("  Group ID = " +
                netPeerGroup.getPeer
GroupID().toString());
        System.out.println("  Peer name = " +
                netPeerGroup.getPeerName());
        System.out.println("  Peer ID = " +

netPeerGroup.getPeerID().toString());
        System.out.println( "Good Bye ....");
        myapp.netPeerGroup.stopApp();
          System.exit(0);
    }

    private void startJxta() {
        try {
          netPeerGroup = PeerGroupFactory.
newNetPeerGroup();
        } catch (PeerGroupException e) {
          System.out.println("fatal error :
group creation failure");
          e.printStackTrace();
          System.exit(1);
        }
    }
}
```

public void startJxta() a pour rôle d'initialiser la plate-forme JXTA et de créer le groupe. Cette initialisation est appelée par la ligne netPeerGroup = PeerGroupFactory.newNetPeerGroup(); qui renseigne implicitement les paramètres par défaut. Pour compiler un exemple comme celui-ci, vous devez vous trouver dans le répertoire du fichier, puis utiliser la commande suivante :

```
javac -classpath ..\lib\jxta.jar;..\lib\
log4j.jar;..\lib\beepcore.jar;..\lib
\jxtashell.jar;..\lib\jxtasecurity.jar;..\lib
\cryptix- sn1.jar;..\lib\cryptix32.jar;..
\lib\jxtaptls.jar;..\lib\bcprov-jdk14.jar;
SimpleJxtaApp.java
```

Et l'exécution passera par la commande suivante :

```
java -classpath ..\lib\
jxta.jar;..\lib\log4j.jar;..\lib\beepcore.jar;..
\lib\jxtasecurity.jar;..\lib\cryptix32.jar;..
\lib\cryptix asn1.jar;..\lib\jxtaptls.jar;..\lib
```

```
\bcprov-jdk14.jar;. SimpleJxtaApp
```

Comme d'habitude avec une application JXTA, la première exécution affichera la fenêtre de configuration, elle-même composée de 4 onglets Basic, Advanced , Rendezvous/Relays et Security. Dans le menu Basic, vous devez saisir le nom du réseau, dans le menu Advanced vous devez sélectionner l'adresse IP de la machine à l'aide du menu déroulant, et décocher le point «Enabled» de l'option HTTP Settings. Dans «Security» vous devez donner un nom d'utilisateur et un mot de passe. Ceux-ci vous seront réclamés à chaque exécution, même après avoir configuré l'application.

Et voici ce que donne l'exécution :

```
TP_JXTA\HelloWorld>java
-classpath ..\lib\jxta.jar;..\lib\log4j.jar;..\lib
\beepcore.jar;..\lib\jxtasecurity.jar;..
\lib\cryptix32.jar;..\lib
\cryptix-asn1.jar;..\lib\jxtaptls.jar;..\lib
\bcprov-jdk14.jar;. SimpleJxtaApp
Starting JXTA ....
Hello from JXTA group NetPeerGroup
  Group ID = urn:jxta:jxta-NetGroup
  Peer name = Programmez
  Peer ID = urn:jxta:uuid-59616261646162
614A7874
61503250337FC3E42067724CD890028D
BE9AACDD3F03
Good Bye ....
```

3 Conclusion

JXTA vise l'interopérabilté, l'indépendance, l'ubiquité et la sécurité. Autrement dit, une application P2P ne doit pas s'appuyer que sur le seul protocole TCP, et ne doit pas exister que sous le seul système d'exploitation Windows. Elle doit pouvoir s'affranchir du matériel (PDA, téléphone mobile, serveur, etc.), et se montrer indépendante du langage de programmation.

Avantages :

- interopérabilté, indépendance, ubiquité et sécurité

Inconvénients

- change rapidement, faible documentation surtout hors du cadre Java, pas d'étude concernant ses performances à une grande échelle de déploiement.

Liens utiles

JXTA Bitvault :
http://www.312inc.com./bitvault_network.html

Distribution JXTA :
http://download.jxta.org/easyinstall/install.html

■ Xavier Leclercq
Xavier.Leclercq@programmez.com

Les protocoles supportés par JXTA

1) L'Endpoint Routing Protocol (ERP) : avec celui-ci, il est facile d'obtenir des informations sur le routage d'un pair à un autre.
2) Le Rendezvous Protocol (PDP) : avec celui-ci, vous pourrez propager un message dans un peerGroup.
3) Le PeerResolverProtocol (PRP) : il permet d'envoyer une requête à plusieurs peers à la fois.
4) Le PeerDiscoveryProtocol (PDP) : celui-ci publie et découvre les ressources.
5) Le PeerInformation Protocol (PIP) : il donne des informations sur les autres pairs.
6) Le Pipe BindingProtocol (PBP) : il est capable de maintenir un canal de communication vers un ou plusieurs peers, en vue d'envoyer un message à un seul pair.

P2P sous .Net avec les sockets

« Il n'existe pas, pour l'instant, d'API stable pour .Net spécialement dédiée au P2P, mais cela n'empêchera pas un développeur de programmer son application peer-to-peer en utilisant les sockets ».

1 Comment programmer les sockets

Un socket est une interface logicielle à partir de laquelle un développeur élaborera un service sur base d'un protocole réseau. Cette programmation par socket est universelle et standardisée (on le doit à l'université de Berkeley qui a inventé ce système il y a plus de 20 ans).

Les espaces de noms du framework .net impliqués dans cette programmation des sockets sont :

```
using System.Net;
using System.Net.Sockets;
```

Le peer, qui est à la fois client et serveur, créera un objet socket de la classe du même nom en spécifiant le type d'adresse (Appletalk, Ipx, DecNet, NetBios, Sna, Unix, ou InterNetwork (adresses IP codées sur 4 octets)), le type de socket (Dgram, Raw, Stream) et le protocole utilisé (Icmp, IP, UDP, TCP).

```
Socket sock;
sock = new Socket(AdressFamily.
InterNetwork, SocketType.
Stream, ProtocolType.Tcp)
```

Le programme serveur doit ensuite exécuter Bind en spécifiant l'adresse IP de sa machine ainsi que le numéro de port :

```
n = sock.Bind(new IPEndPoint(IPAddress.Parse
('192.168.0.24'),6000);
```

L'objet IPEndPoint regroupe une adresse IP et un numéro de port et forme un objet dit «de connexion».

Ensuite, le serveur doit se mettre à l'écoute :

```
sock.Listen(1);
```

Et se place en attente d'une demande de connexion :

```
Socket sockClient = sock.Accept();
```

Le mécanisme est alors le suivant : le serveur se bloque, et ne réagit plus tant qu'il ne reçoit pas une demande de connexion d'un client. La demande reçue, le socket sockClient sera utilisé pour communiquer avec le client.

Du côté du client, cette demande a été initiée par une ligne :

Les sockets ont été mis au point par l'université de Berkeley il y a déjà quelques décennies. Les fonctions C écrites à l'époque sont devenues une norme pour la programmation réseau de bas niveau.

```
sock.Connect(newIPEndPoint
(IPAdress.Parse('192.168.0.24'), 6000));
```

Que peut faire le serveur ? Par exemple envoyer des données au client :

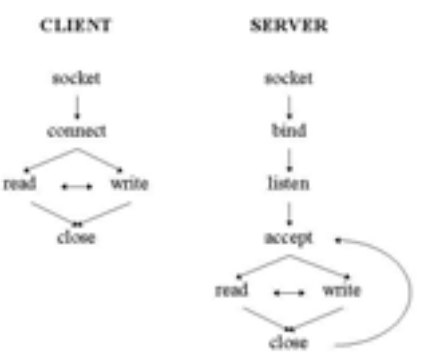
```
string sdonnees = 'mes donnees';
byte[] tb = ASCIIEncoding.
Default.GetBytes(sdonnees);
n = sockClient.Send(tb);
```

Attention, pour que cette chaîne de données soit bien réceptionnée (par le client) il faut qu'il se mette à lire celles-ci :

```
byte[] donnees = new byte[100];
int n = sock.Receive(donnees);
string chainerecue =
 ASCIIEncoding.Default.GetString(donnees);
```

2 Comment expédier un fichier de sauvegarde vers un peer

Mais la programmation d'un logiciel P2P ne s'arrête pas à la programmation des sockets. D'une part nous vous conseillons de travailler avec des règles, car il ne faut pas oublier que vous devez impérativement prévoir le blocage de paquets anormaux. En effet, au départ de la communication, le client demandera au serveur d'établir une connexion (avec un paquet TCP sans données et avec le flag SYN activé). Durant une connexion c'est le seul paquet TCP sans flag ACK qui sera impliqué. Ce type de paquet est toujours envoyé d'un client vers un serveur et jamais l'inverse. Si c'est le cas, alors ce paquet est "anormal" et devra être bloqué. Si les pairs de votre logiciel P2P communiquent sur Internet (ne se limitant pas au seul intranet) c'est important d'y penser.



Résumé du mécanisme de communication par socket.

Ou du moins de penser à mettre au point un mécanisme qui sécuriserait les échanges.

D'autre part, il est bon de prévoir un serveur central. Celui-ci maintiendra à jour les informations à propos des clients qui sont connectés au réseau. C'est-à-dire qu'il pourra vous fournir leurs adresses IP, et la liste des fichiers qu'ils possèdent (par exemple dans le cadre d'une application de sauvegarde P2P). Ce serveur central pourra aussi permettre les recherches dans le cas d'une demande de restauration d'un fichier.

Ceci dit, voici un extrait de code qui résume un peu ce que nous avons expliqué sur les sockets et qui permet d'envoyer un fichier à un peer. L'adresse IP est ici résolue par DNS, car c'est toujours plus simple de travailler avec un nom plutôt qu'avec une adresse IP :

```
public void Envoyer_Fichier_Peer
(String Nom_Du_Serveur,
string Fichier_Distant, string Fichier_Local)
  {
   try
   {
    TcpClient tcpc = new TcpClient();
    Byte[] read = new Byte[1024];
    int port = 6000;
     IPHostEntry
IPHost = Dns.Resolve(Nom_Du_Serveur);
    string []aliases = IPHost.Aliases;
    IPAddress[] addr = IPHost.AddressList;

    IPEndPoint ep =
new IPEndPoint(addr[0], port);
    tcpc.Connect(ep);

    Stream s = tcpc.GetStream();

Byte[] b = Encoding.ASCII.GetBytes
(Fichier_Distant.ToCharArray());
    s.Write( b, 0,  b.Length );
    int bytes;
    FileStream fs = new FileStream
(Fichier_Local, FileMode.OpenOrCreate);
    BinaryWriter w = new BinaryWriter(fs);

    while( (bytes = s.Read
(read, 0, read.Length)) != 0)
    {
     w.Write(read, 0, bytes);
     read = new Byte[1024];
    }

    tcpc.Close();
    w.Close();
    fs.Close();
   }
   catch(Exception ex)
   {
    throw new Exception(ex.ToString());
   }
  }
```

■ Xavier Leclercq
Xavier.Leclercq@programmez.com

Finir en Inde... le truisme de la presse informatique ?

Ce n'est pas pour me déplaire : l'article de Marie Varandat, dans Programmez! d'avril 2006, a le mérite de ne pas laisser indifférent. Je vous livre le chapô de l'article : "Avez-vous le bon profil ? Développeurs ! Bougez pour ne pas finir en Inde". Aussi provocateur et intéressant soit-il, il n'en demeure pas moins légèrement caricatural et relaie des propos assez convenus.

Premier postulat : le développeur en France va disparaître.
Je cite : "Ce serait toutefois se leurrer que d'imputer la disparition du développeur en France à l'offshore uniquement". Sur quels éléments statistiques se fonde l'assertion ? Mystère et boule de gomme. L'élément d'explication esquissé n'en demeure pas moins des plus intéressants. La modélisation UML et les méthodes de travail en mode projet conjuguées à la délocalisation de l'économie accélèrent l'écroulement de la demande. Elle évoque ensuite la saturation de l'offre. Sur ces deux points, les faits semblent montrer le contraire.
Sur le site CadresOnLine, selon 01 Informatique du 14 avril, la demande de développeurs .Net n'a jamais été aussi haute. Rappelons que, selon le Munci, l'année 2005 a été marquée par 6000 à 8000 créations d'emploi nettes dans le secteur de l'informatique. Dans LMI du 24 mars, Hélène Truffaut reprend une analyse de Bernard Riquier Conseil qui semble montrer que la reprise est là. Surtout en matière de développeurs. Or si la demande reprend, c'est que l'offre ne convient pas. L'émergence du logiciel libre et la demande en terme de technologies Internet, du fait de la migration des applications de type client lourd, amènent tout naturellement les SSII à proposer des offres très intéressantes sur la base des CMS ou gestionnaires de contenus. A Niort, c'est ainsi que la société Cyberscope axe son offre de sites autour de Typo 3. L'utilisation de ces plates-formes minimise le coût global pour les entreprises utilisatrices et permet de maintenir les marges pour les prestataires. Elle éloigne le spectre de l'offshore.

Deuxième postulat : le salut de l'informaticien serait d'être un médiateur. *Dans l'article, 7 métiers sont mis en exergue : architecte, chef de projet, web designer, rssi, testeur, consultant-expert. A l'expérience des faits, rappelons que ce sont les chefs de projet qui, dans ces dernières années, ont le plus souffert. Les entreprises ne savent pas bien quoi faire de profils généralistes et peu techniques. Là où elles dégraissent, ce sont les profils les plus techniques qui restent. Pour qu'ils puissent rester en poste, leur tâche consiste à proposer des économies de coûts qui leur permettent de justifier leur poste (Voir la guerre des places de Vincent de Gauléjac). Par ailleurs, la conception de sites, la sécurité, la gestion de projets, le testing, la qualité, la réalisation d'infrastructures sont intégrées aux formations relatives au développement. Autrement dit, un bon développeur peut devenir un bon chef de projet ou un bon web designer. L'inverse est complètement faux ! Quand à l'autisme évoqué par Michel Lara d'IBM concernant les jeunes de l'EPITA, il est davantage lié à des processus de désocialisation avancée, liés à la pratique des jeux vidéo et de MSN. Certes, le phénomène est inquiétant. Pour autant, il relève d'une tout autre problématique. Ne mélangeons pas tout.*

Michel Lara et le Gartner group sont-ils les mieux placés pour parler de notre secteur ? *Même s'ils ont le droit imprescriptible et inaliénable de s'exprimer sur les sujets relatifs à l'informatique, rappelons que l'emploi en France est assuré par les entreprises de moins de 50 salariés. Le tropisme qui est naturellement le leur, compte tenu de leur activité, les autorise-t-il à généraliser leur vision très globalisante ? L'émergence d'UML, de l'objet, d'ITIL, des technologies de type framework, l'hypertechnicité "gadgétique" sont produits par la volonté d'industrialiser et de tayloriser le développement. Il s'agit sans doute d'une manière "élégante" de faire le ménage dans ce secteur. De fait, nous devons nous adapter à cette nouvelle donne. Pour qu'elle ne soit pas subie, il faut la comprendre.*
En nous appuyant sur le logiciel libre, les méthodes agiles, nous pouvons relever le défi économique de l'offshore et y faire face dans des conditions économiques favorables aux métiers du développement logiciel... en France. La vision poussée par l'article s'inscrit dans le discours dominant qui tend à nous expliquer la nécessité de la spécialisation, s'appuyant sur la théorie de l'avantage comparatif. A cette vision idéologique, nous devons pousser un vision pragmatique qui, seule, peut nous permettre de pérenniser l'emploi "régional".

Denis Szalkowski
Consultant
Site : http://www.dsfc.net/
Mon blog : http://www.dsfc.net/dotclear/

Java 6 et Java EE5

Simplicité - Productivité - Vélocité

Le monde Java subit la concurrence de la plate- forme .Net de Microsoft, qui s'est imposée comme une réelle alternative. Sun et les promoteurs de Java se devaient de réagir. Ils sortent un Java qui répond aux attentes : plus simple, plus productif, plus véloce.

.Net fait quasiment jeu égal avec Java / J2EE ! Selon Forrester Research, 44 % des entreprises utilisaient ou utiliseraient .Net pour les applications clientes contre 35% pour Java/J2EE (étude de fin 2005, sur un échantillon de 869 entreprises internationales).

Par contre, pour le langage utilisé par ces mêmes applications, Java est devant, avec 39 % contre 37% pour VB et 19% pour C#.

Plus que jamais, Java et .Net se partagent et se partageront le marché applicatif. Cependant, si cette donne est valable pour les nouveaux projets, Java a l'avantage de posséder un existant (un legacy) ancien et en nombre. De son côté, .Net fait face à un existant VB 6 non négligeable et persistant.

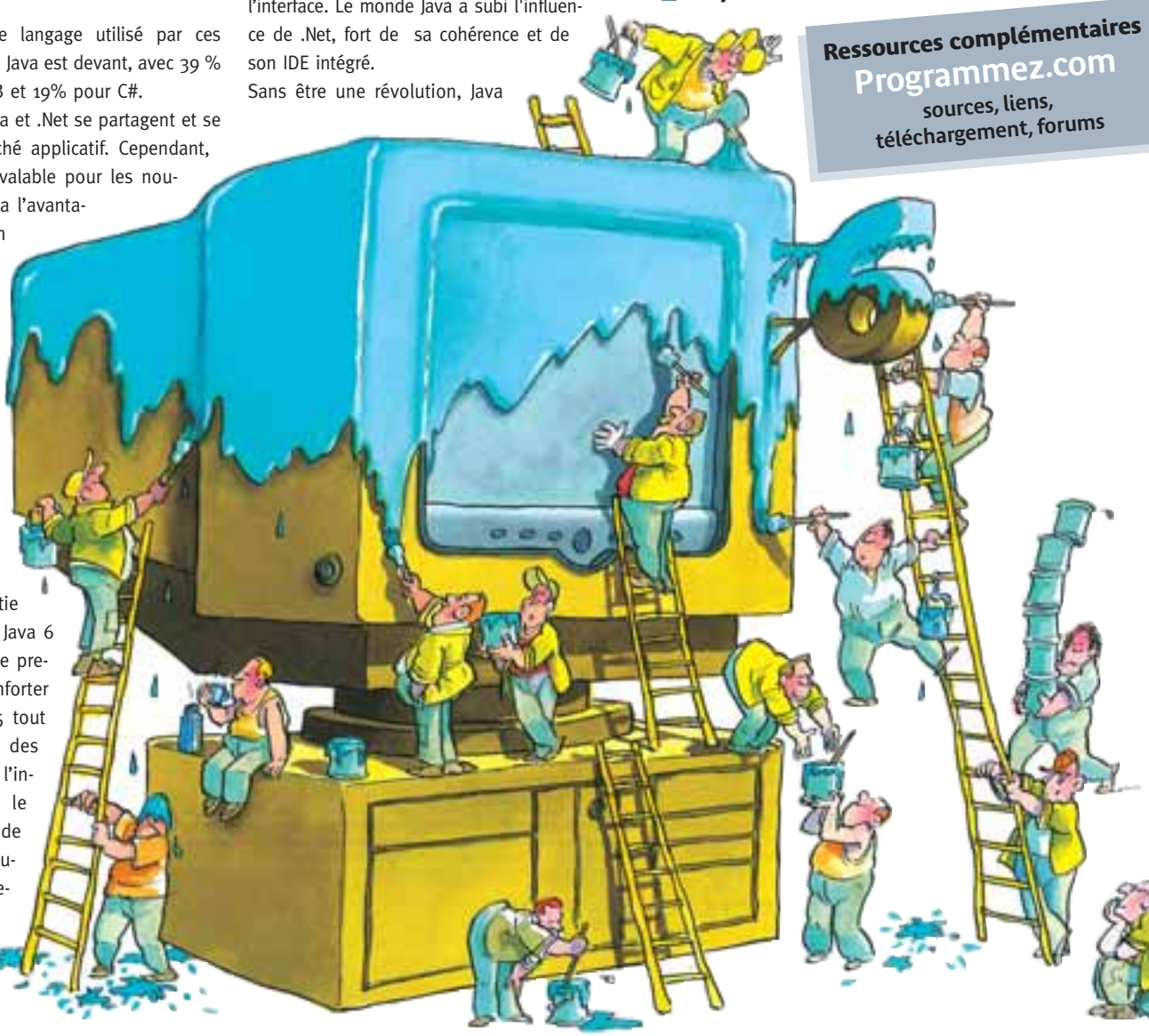
Java 2006 : un excellent cru

Fait exceptionnel, 2006 marquera, sauf retard, la sortie de deux versions : Java 6 et Java EE 5. Pour le premier, il s'agit de conforter les bases de Java 5 tout en proposant des aspects natifs dans l'interface. Pour le second, il s'agit de simplifier l'architecture et le développement d'applications entreprises. J2EE traînait depuis trop longtemps des lourdeurs dans l'architecture et surtout dans les EJB. Le fait que Java 6 s'intègre toujours plus au système sur lequel tourne l'application, doit fournir une nouvelle dimension des applications Java, plus orientées desktop, avec un soupçon de fonctions natives dans l'interface. Le monde Java a subi l'influence de .Net, fort de sa cohérence et de son IDE intégré.

Sans être une révolution, Java réussit à rebondir sur les bases solides de Java 5 – Tiger, tout en apportant des nouveautés intéressantes. Dans ce dossier, nous allons vous montrer l'ensemble des nouveautés dans Java 6, Java EE 5 et EJB 3.

François Tonic

Ressources complémentaires
Programmez.com
sources, liens, téléchargement, forums



Java 2006 : un agenda chargé !

Après 2005, année Microsoft et .Net 2, 2006 sera-t-elle l'année Java ? On peut le penser avec l'arrivée durant les prochains mois d'évolutions majeures des plates-formes Java. Avant d'aller plus avant dans la mise en pratique des nouveautés 2006, revenons sur l'agenda, les évolutions, le futur.

Cela fait déjà plus d'un an que l'on titrait dans Programmez ! sur Java 5 et le futur du langage, notamment avec Java 6 et le très lointain Java 7. En 2006, nous aurons droit à deux nouveaux Java : Java 6 et Java EE 5. D'ores et déjà les premiers éléments de Java 7 se dévoilent même si cette édition ne sortira pas avant courant 2008.

1 Java 6 alias Mustang – automne 2006

Après Java 5 – Tiger, Sun reste dans le domaine animalier, avec Mustang pour Java 6. Tiger avait été à plusieurs reprises repoussé et a finalement tenu ses promesses sur les fonctionnalités. Java 5 donnait un sérieux coup de fouet dans le modèle de programmation grâce aux génériques, aux annotations, etc. Java 6 reprend les bases de Java 5 sans apporter de grandes nouveautés de programmation. Pour Sun, Mustang doit répondre aux critères suivants : compatibilité, qualité et stabilité. Sur la qualité, les mécanismes et fonctions de monitoring ont été améliorés dans JMX et la console Jconsole.

Les **Web Services** constituent le gros morceau nouveauté avec une intégration directement dans la JDK, facilitant ainsi la création et la consommation de services web. Les fonctions de base sont ainsi présentes. Pour faciliter cela, Java 6 incorpore un mini serveur web. Outre JAX, les WS* seront incorporés mais, sur ce point, il faudra attendre encore un peu avant de savoir quelles seront les spécifications intégrées dans la JDK (selon l'avancée des travaux). L'interopérabilité avec les services Web du monde Microsoft n'est pas oubliée (voir encadré).

Surtout, Sun pallie une lacune de Java 5 sur les **annotations**. Java 6 arrivera avec des annotations prédéfinies, pour rapidement les utiliser dans son code. Les changements les plus importants se situent sur l'utilisation de langages scripts. Ainsi, il devient possible d'utiliser JavaScript dans la JVM, via Rhino. Rhino est un moteur JavaScript initialement conçu par Mozilla et qui a été modifié pour s'intégrer à Mustang. La JSR 223, définissant le Scripting, est supportée dans Java 6 et autorise donc, théoriquement, l'usage de tout langage script comme Groovy, PHP, JavaScript, etc. Java 6 s'introduit ainsi dans le concept Web 2.0 et surtout dans les applications internet riches. Nous ne sommes qu'au début de cette intégration. Mustang met l'accent sur JavaScript pour le moment.

Le **poste de travail** constitue la réelle nouveauté de Java 6 comme vous le verrez dans les articles pratiques de ce dossier. Swing possède des fonctions d'anti aliasing, un meilleur support des interfaces et préférences des systèmes, la disponibilité immédiate d'un écran de démarrage, l'apparition du système de Tray. Le mot d'ordre est « integration desktop ». Cela signifie que l'application Java utilisera des fonctions et API sur le système sur lequel elle tourne. On touche donc au principe d'un code unique tournant partout, mais est-ce le plus important ? L'important est de disposer d'une application mieux intégrée à un système, pouvant tirer parti de celui-ci.

Côté **sécurité**, Mustang intègre GSS / Kerberos pour les services d'authentification, accès aux serveurs LDAP pour l'authentification des utilisateurs, un accès simplifiée aux services de cryptographie de Windows ou aux PKI. Pour les signatures numériques, on bénéficie de XML-DSIG.

Les **données** ne sont pas oubliées. Mustang intègre, par défaut, JBDC 4.0. L'un des objectifs de cette version est de fournir aux développeurs une meilleure productivité dans leur travail avec SQL et les données. Pour faciliter la connexion, JDBC s'occupera de trouver le bon driver pour se brancher à la base de données. Cela simplifie grandement l'implémentation. On bénéficie aussi du support XML pour SQL : 2003, d'amélioration sur les objets Blob. L'apparition du XML permettra d'utiliser et de manipuler des données XML stockées dans un SGBD.

Les JSR de Java 6 à retenir

269	Pluggable Annotation Processing API. Le but est de créer rapidement des annotations.
199	Java Compiler API. Donne les descriptions des fonctions du Java Language Compiler.
223	Définit l'usage des langages script dans Java
224	Spécification de Jax-WS 2.0
222	Spécification de JaxB 2.0. Architecture du XML binding.
221	Spécification de JDBC 4.0.

Une adoption plus rapide ?

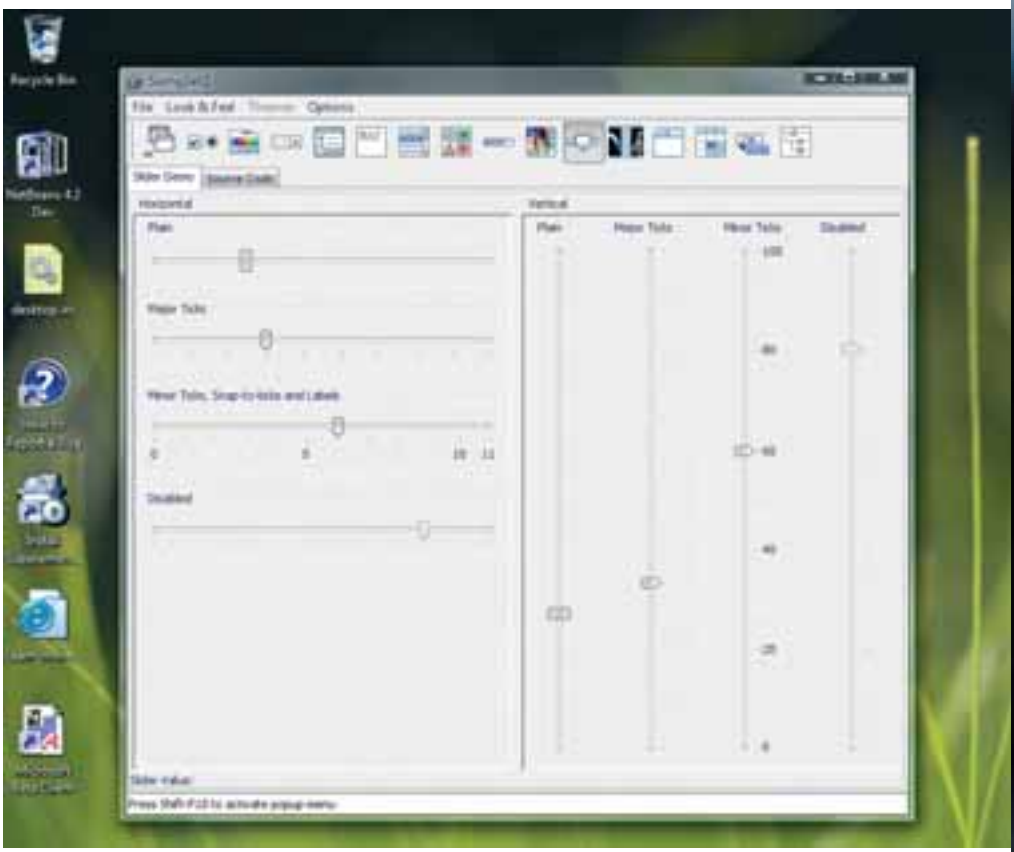
Une des clés du succès de Java 6 dépendra de la rapidité d'implémentation de cette version dans les outils de développeurs, de manière standard. NetBeans le supportera en standard sans doute avec la version 6.0 prévue à la fin de l'année. NetBeans 5.5 apportera le support de Java EE 5. Eclipse devrait supporter, par

défaut, Java 6 lui aussi rapidement. Sur Java EE 5 l'enjeu est le même, avoir une adoption « rapide », car J2EE 1.4 avait eu une prise en compte dans les outils assez laborieuse. Sur ce point, ObjectWeb supporte déjà EJB 3 et le projet Glassfish de Sun propose une implémentation de référence de Java EE 5.

2 Java EE 5 – printemps 2006

Attendue avec impatience, Java EE 5 (ex J2EE) sera disponible dans quelques semaines. Deux grandes nouveautés se démarquent : EJB 3 et JavaServer Faces, sans oublier un meilleur support des services web. Le socle est celui de Java 5. On notera que les annotations sont largement utilisées dans Java EE 5, notamment pour le déploiement qui s'en trouve grandement simplifié (la description devenant facultative), pareillement pour la configuration. Outre la simplification d'une partie de l'administration par l'annotation, le but de Java EE 5 est de réduire le code Java et XML nécessaire, bref, faire moins de déclarations. Cette réduction est possible grâce à l'utilisation du modèle POJO. Jusqu'à présent, il n'était pas forcément simple d'accéder aux ressources d'une application J2EE. Avec la version 5, il sera possible d'injecter des ressources nécessaires à un endroit précis. Et le debug sera rendu plus simple.

Une importante avancée est faite sur les mécanismes de persistance. Jusqu'à présent, une guerre se déroulait entre, principalement, JDO et Hibernate. Avec Java EE 5, Sun introduit une persistance créée par l'équipe EJB 3, capable de fonctionner en dehors des EJB et utilisable en Java 6, applications desktop ou web, avec Swing, etc. Cette technique repose sur le principe POJO, propose un support natif des requêtes, qui sont par défaut polymorphiques. Sur le service web, on retrouve les mêmes avancées que dans Java 6 avec la présence de Jax-WS 2. Une autre nouveauté est l'implémentation par défaut des JavaServer Faces 1.2. En complément, Java EE 5 inclut par défaut Java Standard Tag Library. JSF 1.2 reprend et améliore, légèrement, la v1.1. Il s'agit essentiellement d'aligner les technologies JSP et JSF pour mieux les utiliser et simplifier le code. Les véritables nouveautés seront dans la future v2 des JSF.



3 Java 7, alias Dolphin – 2008

En toute logique, Java 7 ne sortira pas avant 2008 si on suit un cycle de développement de 18 mois. Aucun JSR n'est encore été formé. Quatre grandes tendances se dégagent :

- intégration d'une syntaxe XML native dans Java
- amélioration de la productivité du développeur
- redéfinition de la notion de module (en référence aux travaux du JSR 277)
- en liaison avec la notion de module, le fichier JAR évoluera et devrait apparaître le module avec dépendances.

Si le Web Service va continuer à évoluer dans son intégration et support au sein de Java, au niveau architecture, le travail envisagé sur les JAR et les modules doit permettre de créer des applications plus modulaires. Il est possible que de nouveaux objets d'interfaces apparaissent. JSR 277 vise à supprimer le JAR et à mettre en place des Java modules et un repository.JSR 277 vise pour le moment uniquement Java SE mais devrait en toute logique être pris en compte dans un futur Java EE. Il doit permettre d'unifier le modèle de déploiement des extensions et applications Java. Le scripting déjà introduit dans Java 6 sera bien plus évolué dans Dolphin (notamment grâce à la présence d'un shell). Il nécessitera moins de code pour utiliser du script dans son application Java. Les toutes premières builds seront disponibles courant de l'été.

4 Java EE 6 – 2008 / 2009

Aucune information.

Le projet Tango

Comment assurer l'interopérabilité des services d'entreprise entre Java Web Services et Microsoft Windows Communication Foundation ? Chez Sun, ce travail est assuré par le projet Tango qui implémente de nombreuses spécifications de WS*. Le projet s'articule sur plusieurs pôles : la sécurité, un Core XML, la qualité, le messaging et le metadata. Il est prévu que l'outil d'interopérabilité soit disponible durant la prochaine JavaOne.

■ **François Tonic**

Choisissez votre propre voie

Les Outils de Développement Intel® convertissent la puissance Multi Noyaux en Performances.

Les **Intel® Threading Tools** permettent de localiser des erreurs de fils d'exécution et d'optimiser les applications multi filaires pour obtenir les meilleures performances possibles.

Lec **Compilateurs Intel® C++ et Fortran** rendent les performances des applications multi filaires très performantes et améliorent la productivité des développeurs.

Les **Analyseurs Intel® VTune™** saisissent, analysent et affichent les données d'exécution pour identifier les goulets d'étranglement des applications multi filaires.

L'**Intel® Integrated Performance Primitives** est une bibliothèque logicielle qui intègre une gamme étendue de fonctionnalités d'imagerie et de traitement du signal.

Economisez jusqu'à 120€	Prix €[ht]
Intel® Threading Tools 2.2	594,00
Intel® C++ Compiler 9.1 pour Windows	339,00
Intel® Visual Fortran Compiler 9.1 Standard Edition pour Windows	424,00
Intel® Visual Fortran Compiler 9.1 Professional Edition pour Windows	1189,00
Intel® VTune Performance Analyzer 8.0	594,00
Intel® Integrated Performance Primitives 5.0 pour Windows	169,00

microsigma

Pour en savoir plus, contactez Valérie ou Frédéric au 01 30 82 04 54, envoyez-nous un email à intel@microsigma.fr ou allez sur notre site à www.microsigma.fr/intel

intel Software

*2006 Intel Corporation, Intel, le logo Intel, Pentium, Itanium, Intel Xeon et VTune sont des marques ou marques déposées d'Intel Corporation ou de ses filiales aux États-Unis et dans d'autres pays. Les autres noms et marques peuvent être réclamées comme appartenantes à d'autres.

Interview

La stratégie de Sun « L'approche communautaire est l'approche de notre temps »

Emmanuel Obadia, après une longue carrière en France, notamment chez Lotus et IBM, est directeur, au siège de Sun, des communautés de développeurs. Il nous apporte sa vision de l'évolution de Sun et de Java, et de leur impact sur les développeurs.

Programmez : Quelle est la stratégie de Sun concernant ses outils de développement, alors qu'ils sont désormais ou open source, ou gratuits ?

E. Obadia : Comme Jonathan Schwartz le rappelle à beaucoup d'occasions "Developers don't buy things, they join things". Ils rejoignent les communautés où les équipes de développeurs épars, animés d'intérêts semblables peuvent collaborer, malgré de grandes distances, autour de problématiques de développement connexes. Au sein de ces communautés, les attentes sont que le logiciel doit être librement accessible. Le marché des outils de développement a nettement changé au cours des 5 dernières années, notamment l'utilisateur final a plus de voix et d'influence dans la structure de prix des outils logiciels. Le modèle économique traditionnel d'achat et de licensing du logiciel est impacté de manière significative par le mouvement open source, au point que les fournisseurs de logiciels revoient leurs politiques. Sun reconnaît ce changement de dynamique en offrant ses outils de développement gratuitement et en les plaçant pour partie en open source.

A long terme, les sources de revenus pour Sun en matière d'outils logiciels, comme d'autres fournisseurs de logiciels, proviendront du support, des services, de la formation et de l'éducation. A ce titre, aucune entreprise cliente utilisant un outil gratuit n'est disposée à couvrir la distance avec un produit sans support et/ou un niveau approprié de formation pour maximiser leur utilisation de l'outil.

L'avènement du collaboratif et du partage ne peut plus être contré dans l'économie actuelle. Tout acteur pensant préserver cet ancien modèle de distribution logicielle basé sur l'achat de licence est en train, dans une large mesure, de se battre contre la gravité terrestre. Notre ambition consiste donc à accueillir le plus grand nombre de développeurs au sein de nos communautés (Sun Developer Network, Java.net, Open Solaris, NetBeans, ...), les voir adopter nos outils gratuits, contribuer et bénéficier de nos apports à l'open source. Cela induira encore plus d'innovation autour de sujets clé et d'opportunités de business pour l'ensemble de la communauté.

Emmanuel Obadia
World-wide Field Marketing Director,
Demand Generation chez Sun.

> *"L'avènement du collaboratif et du partage ne peut plus être contré dans l'économie actuelle. Tout acteur pensant préserver cet ancien modèle de distribution logicielle basé sur l'achat de licence est en train, dans une large mesure, de se battre contre la gravité terrestre."*

Programmez : Quel est le soutien de Sun à Netbeans ?

E. Obadia : NetBeans existe depuis longtemps, et Sun a saisi l'occasion d'ajouter cette plate-forme open source à son offre logicielle.

Bien que l'on puisse considérer Sun comme administrateur de la plate-forme NetBeans, le produit appartient vraiment aux développeurs qui l'emploient en leur permettant d'armer la puissance de l'outil ou de construire leurs propres plug-ins au-dessus

FORMATIONS EN ALTERNANCE

Acquérez une expérience en vous formant DE BAC À BAC +5

Analyste développeur / Technicien Supérieur Réseaux Télécoms Concepteur Développeur de Systèmes d'information / Architecte Réseaux Multimedia / Chef de projet / Responsable sécurité des Systèmes d'Information / Développeur de Jeux Vidéo

Cisco Systems Networking Academy · Microsoft | IT Academy Program · Pearson VUE · Novell · Oracle · NETASQ

iie

institut informatique & entreprise

Plus d'infos au 03 27 284 363

e-mail : iie@valenciennes.cci.fr

Cette école bénéficie du soutien de l'Union Européenne dans le cadre du FSE et du FEDER

CCI du Valenciennois GROUPE FORMATION

pour répondre aux divers besoins métier. En fait, Sun construit ses propres outils à partir de la plate-forme NetBeans : Sun Java Studio Enterprise et Sun Java Studio Creator (pour adresser le marché de Java) et Sun Studio (pour adresser le marché d'outils natifs - C/C++, FORTRAN...); qui sont maintenant tous disponibles gratuitement. Alors que Sun offre la plate-forme NetBeans comme outil open source gratuitement, Sun propose une offre commerciale de support pour les besoins des développeurs.

Programmez : **Quelle est l'évolution de la communauté des développeurs java ? Comment voyez-vous son futur ?**

E. Obadia : La communauté Java est riche de développeurs d'horizons aussi différents que Java est mu par la mantra : coder une fois, exécuter partout (write once, run everywhere). L'ensemble des "Java enabled devices" a dépassé les 3 milliards d'unités; par exemple, le nombre des téléphones portables Java a passé la barre du milliard d'unités dans le monde, le nombre de PC équipés de la machine virtuelle Java celle des 700 millions d'unités, sans mentionner les serveurs J2EE, les Java Cards et les appareils en tout genre (Set Top Box, PDA, Consoles de Jeu, Automobiles... jusqu'au Rover qui a exploré Mars !) opérant la plate-forme Java.

Pour répondre plus directement, l'ubiquité est probablement le maître mot de cette communauté et le nombre de développeurs Java recensés est au delà de 4,5 millions. Vaste question donc.

> *“Coder une fois exécuter partout”*

Je vois le futur de cette communauté rassemblant toutes les énergies et les idées autour de cette idée communautaire de partage pour le bénéfice du plus grand nombre: les développeurs Java Oracle, SAP (avec NetWeaver), ou ceux qui s'efforcent de produire les meilleurs jeux vidéos, etc. Le partage, l'ouverture, la libre contribution et l'autorégulation sont autant d'attributs qui ont forgé le succès de Java depuis 1995 et qui ne s'est jamais démenti. Ce succès et cette expansion s'inscrit dans le mouvement plus général de l'industrie IT, que nous caractérisons chez Sun par le slogan "Share" et que nous décrivons comme l'ère de la participation (Participation Age) que d'autres décrivent dans des concepts forts comme par exemple Web 2.0 (Tim O'Reilly). Ce mouvement et par conséquent cette communauté ne peut que s'amplifier car, comme le disait Victor Hugo "Rien n'a plus de force qu'une idée dont l'heure est venue."

Programmez : **Comment voyez-vous la concurrence de .NET ? N'assiste t-on pas à une déferlante ?**

E. Obadia : Je retourne la question a dessein, comment Microsoft voit la concurrence avec Java ? Nous avons depuis Avril 2004 un accord de partenariat très étroit avec Microsoft et courant sur 10 ans, dans le cadre duquel l'interopérabilité de Java et .net est directement traitée entre Sun et Microsoft. La raison en est simple: l'intérêt des clients. Sun et Microsoft n'ont pu que faire le constat que les entreprises allaient probablement disposer de ces deux plates-formes. D'ailleurs, ce sont les 2 seules plates- formes restantes en terme d'architecture logicielle.

Quand à la portée de chacune d'elles en termes applicatifs, quant au degré d'innovation, au nombre de contributions des développeurs, ouverture ... et nombre d'appareils supportant chacune d'elles, je vous renvoie à la fin de ma réponse ci-dessus. Java apparaît largement comme la plate forme dominante dans tous les compartiments en forte croissance car l'approche communautaire est l'approche de notre temps.

■ **Propos recueillis par Jean Kaminsky**

Compuware joue la proximité

Si l'éditeur d'outils de qualité comme DevPartner ou OptimaleJ regarde les nouveaux Java, Compuware joue sur le calendrier pour implémenter les nouveautés dans ses outils. « Sur Java 6, on ne l'a pas encore expérimenté. Par contre, on travaille sur Java EE 5 et les EJB 3, qui sont très intéressants. On expérimente et on réalise des tests d'intégration, notamment sur les POJO, dans notre outil OptimaleJ. On va vers une simplification du modèle de développement et vers une meilleure cohérence entre UML et Java. Les annotations sont plus qu'une révolution. Java EE 5 va dans le bon sens. Sur notre outil DevPartner pour Java, Java 6 / Java EE 5, il est peu impacté, l'outil fait avant tout de l'analyse de code.

Il y aura sans doute quelques aménagements à réaliser. Dans le monde Java, la variété des choix prime. Mais on finit par se dire que si on a une bonne solution, cela laisse émerger un standard, c'est peut-être le cas avec les EJB 3. Cependant, maintenant, il faut lire les spécifications, regarder comment cela va réellement s'implémenter dans les outils. Entre les promesses et le marché, il peut y avoir une différence. Sur les EJB 3, il semble se dégager un consensus », précise Guy Cartigny (responsable technique). Cependant, les risques d'une dérive et que chaque solution fasse sa propre implémentation avec des rajouts spécifiques existent. « EJB QL était plein de promesses, malheureusement, tout le monde a fait son « bricolage ». conclut Guy Cartigny.

■ **F.T.**

Avis d'experts

BEA se met aux prochains Java

« Java EE 5 constitue une spécification importante, dans la continuité de Java 5 – Tiger. Sans Java 5, on n'aurait pas les generics ou les annotations. On y trouve aussi l'utilisation des POJO. Tout cela simplifie l'ensemble de Java EE et le développement » précise Alexandre Vasseur (senior System Engineer, BEA). BEA annonce que WebLogic Server utilisant Java EE 5 sortira sans doute à la fin de l'année. Les tests sont déjà en cours. EJB 3 est l'autre grand morceau du prochain Java entreprise. « Sur les EJB 3, on propose déjà une pré-version sur la partie persistance de donnée avec Kodo de SolarMetric ». L'éditeur prépare le projet JPA, un projet Open Source s'appuyant largement sur Kodo, supportant les EJB 3. Au-delà de ce futur outil, BEA réaffirme son intérêt pour Workshop. « Workshop reste un élément central pour nous. Il s'appuie sur Eclipse et la version 3, sortie récemment, supporte déjà les EJB 3 et propose une approche bottom up. Il permet de générer les composants EJB 3 d'un projet » poursuit Alexandre Vasseur. Par contre, sur l'intérêt des clients, utilisateurs à passer à Java EE 5, il y a un fossé qu'ils ne franchissent pas, « c'est encore un peu loin pour eux. Ils se tiennent au courant, mais attendent les spécifications, les certifications. On ne migre pas s'il n'y a pas de pertinence » ajoute Alexandre Vasseur. Un des points « faibles » cités par l'éditeur est le manque de spécifications sur la montée en charge, le clustering, la haute disponibilité, mais finalement « est-ce que l'on a besoin de standardiser cela dans les spécifications » analyse M. Vasseur. Enfin sur Java 6, BEA mettra à jour sa JVM Jrockit dans les 90 jours suivant la sortie de la nouvelle JDK. Cependant, de l'aveu même de l'éditeur « Java 6 ne change pas beaucoup de choses, c'est une version assez mineure. » conclut Alexandre Vasseur.

Faciliter le développement est un atout

La Java EE 5 rend, aujourd'hui, plus souples et plus homogènes les évolutions des versions précédentes. Nous ne considérons pas cette version comme révolutionnaire, même s'il est incontestable qu'elle apporte beaucoup de nouvelles fonctionnalités et de facilités, par exemple au niveau de JSTL. Il s'agit plus, à notre sens, d'une intégration rationnelle et mature des technologies. Cette intégration permet notamment de rendre compatibles les technologies JSP et JSF. Les nouvelles API JAX-WS 2.0 et JAXB 2.0 sont pour nous l'évolution majeure en termes de support, de protocoles, de services et de souplesse. Tout ce qui peut faciliter le développement est un atout pour JDEO puisque nous prenons en charge pour nos clients l'implémentation des composants applicatifs et intégrons déjà cette technologie. Java SE 6 apporte des améliorations et des corrections longtemps attendues au niveau des Awt/Swing. Ce qui nous fait envisager une implémentation technique de certains de nos composants visuels autour de Java. La version précédente 5 ou 1.5 apportait déjà un gain important au niveau des performances et des nouveautés sur le langage.

Vincent LEMOINE
Président JDEO Solutions

Interopérabilité & interface

Java 6, dont le nom de code est Mustang, est défini par le JSR 270. Les bénéfices apportés sont multiples : interface graphique, services Web, meilleur support JDBC, management de la JVM ou encore sécurité. Swing bénéficie d'un ensemble d'améliorations allant du drag and drop au rendering des composants graphiques nativement et d'une plus grande intégration avec le système d'exploitation. Cela réduira la différence en terme d'expérience utilisateur entre les applications Java et les applications natives.
Exposer un service Web intéropérable depuis le monde C# va devenir encore plus simple grâce aux annotations. Cette simplicité va permettre aux mondes Java et C# de communiquer entre eux facilement. JDBC va passer à la version 4 et donc supporter les types de données XML, BLOBs (Binary Large OBjects) et CLOBs (Character Large Objects). Les blobs sont utilisés pour sauver des données binaires (images, exécutables, jars), et les clobs sont utilisés pour stocker de grandes quantités de texte. De plus, intégrer des requêtes SQL sera simplifié grâce aux annotations - @Query(sql="select * from mytable") - ainsi que la gestion des connections.
La possibilité sera offerte d'intégrer des outils de développement dans la machine virtuelle. Par exemple, il sera possible d'intégrer un compilateur natif java. Des outils de monitoring (analyse mémoire et analyse post-mortem) pouvant être attachés à un processus en cours d'exécution seront aussi livrés en standard. Mustang devrait donc améliorer l'utilisabilité des GUI Java grâce aux améliorations de Swing. La productivité des développeurs au niveau des serveurs sera accrue avec l'émergence d'outils de développement sophistiqués. Aussi, les services Web sécurisés et intégrés aux services d'infrastructures du monde professionnel feront de Java une technologie de choix pour les problématiques d'intégration du monde professionnel actuel.

Sébastien Marc
Development Manager

Java EE 5 : vers la simplification

J2EE est une technologie trop complexe ! Que de fois avons nous entendu cette remarque désabusée ? Et de fait, J2EE avec ses innombrables API, contrats, outils de développement et modèles de conception est complexe. Pourtant en 2006, J2EE reste une technologie incontournable dès lors qu'il s'agit de construire des applications d'entreprise où les exigences, en termes de montée en charge, de sécurité, de maintenabilité et de performances sont déterminantes.

Conscient que le ticket d'entrée reste élevé pour un développeur néophyte, Sun a choisi pour la nouvelle édition de sa plate-forme Java d'entreprise de porter l'accent sur la simplicité. A ce titre, 3 nouveautés dont nous parlerons ci-dessous, jouent un rôle déterminant : l'introduction des EJB 3.0, munis d'un nouveau modèle de persistance, un modèle de composants web JSF 1.2 et une mise en œuvre simplifiée des web services avec JAX-WS 2.0. Le liant de toutes ces technologies réside dans l'usage systématique des annotations Java 5. Pour marquer le coup, la plate-forme change de nom : nous passons ainsi directement de J2EE 1.4 à Java Enterprise Edition 5.

Les nouveautés

Un usage systématique des annotations J2SE 5

L'une des sources aisément identifiable de complexité de J2EE est la pléthore de descripteurs de déploiement qui parsèment le code d'une application. Rappelons ici qu'il s'agit de fichiers XML contenant les métadonnées nécessaires au déploiement des composants constituant une application. Java EE 5 prend le parti de rendre désormais facultatifs ces fichiers XML, pour les remplacer par les annotations introduites dans l'édition standard de Java 5. Qu'est ce qu'une annotation ? Pour faire simple : une annotation est un nouveau type de modificateur d'une déclaration (de classe, de méthode ou d'attribut) au même titre par exemple que les modificateurs *transient, final, public* ou la balise JavaDoc *@deprecated*. Elle modifie la manière dont le code Java est traité par des outils, des librairies ou un conteneur. Pour remplacer les descripteurs de déploiement, Java EE 5 définit une panoplie d'annotations standard pour tous les types de composants : EJB, Servlets, WebServices, Filtres, Intercepteurs, etc. Voici un exemple de classe et d'attribut annotés :

```
@Resource(name=
"myDB", type=javax.sql.DataSource)
@Stateful public class
ShoppingCartBean implements ShoppingCart
{
    @Resource SessionContext ctx;

    public Collection startToShop
(String productName) {
        ...
        DataSource productDB
= (DataSource)ctx.lookup("myDB");
        Connection conn
= myDB.getConnection():
        ...
    }
    ...
}
```

Ces trois annotations déclarent respectivement la classe ShoppingCartBean comme un EJB session avec état, une source de données nommée myDB ainsi qu'un contexte d'EJB ctx. Cet exemple illustre deux principes fondamentaux des annotations dans Java EE 5. **L'injection de dépendance** et la **configuration par exceptions.** Pour le premier, il s'agit d'un principe de conception bien connu des utilisateurs de Spring, qui consiste à déléguer l'instanciation des objets à une classe tierce, supprimant ainsi toute dépendance mutuelle entre implémentations. En l'occurrence, dans notre exemple, le conteneur se chargera d'instancier le contexte ctx ainsi qu'une DataSource nommée myDB. Plus de fastidieux lookups JNDI ! Le second consiste à faire en sorte que le conteneur fournisse, chaque fois que possible, des valeurs par défaut convenables pour toutes les ressources. Le nombre d'annotations est ainsi réduit au strict minimum.

Le lecteur familier avec les balises XDoclet y verra là une forte ressemblance. Cette configuration par annotations sacrifie un peu de souplesse de déploiement (il n'y a effectivement plus d'externalisation de la configuration) pour une simplification significative du modèle de programmation.

Les EJB 3.0 et une nouvelle API de persistance

Complexité de déploiement, innombrables subtilités du contrat entité-conteneur, pas d'héritage ni de polymorphisme, performances catastrophiques, voilà quelques-uns des reproches faits aux EJB entités 2.x ! Exit donc les EJB entités ! Désormais, dans Java EE 5, les entités persistantes sont de simples JavaBeans, les fameux POJO (Plain Old Java Objects), déclarés à l'aide d'une annotation *@Entity*. Développée sous les auspices du JSR-220, cette nouvelle spécification EJB 3.0 se veut moins académique que l'ancienne. Pour cela, elle s'inspire des meilleures idées récentes qui ont émergé dans différents contextes : le framework open source Hibernate 3.0, les spécifications JDO 2.0 de la JSR 243 ou des solutions commerciales comme TopLink. En contraste avec les EJB 2.x cette nouvelle API de persistance sera utilisable aussi bien dans un contexte dit « managé », c'est-à-dire au sein d'un serveur d'application, que dans une simple application J2SE autonome.

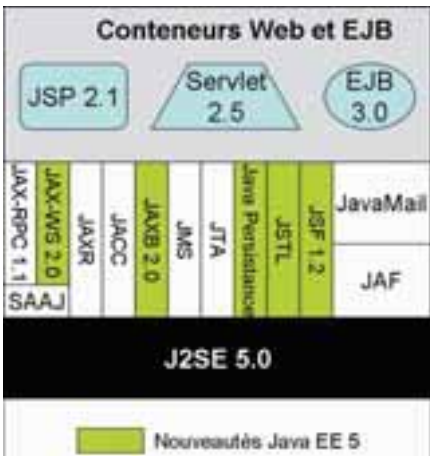
Le concept central est celui d'*EntityManager* auquel est associé un **contexte de persistance** dans lequel des entités peuvent être créées, supprimées, mises à jour ou recherchées. C'est dans ce contexte également qu'est garantie l'équivalence de la notion d'identité des objets Java avec l'identité en base de données. Pour le lecteur familier avec Hibernate, l'EntityManager correspond à la Session Hibernate. Dans le cas d'un contexte managé, le conteneur est responsable de fournir un même EntityManager à

tous les composants impliqués dans une transaction JTA. La durée d'un contexte de persistance peut coïncider avec la durée d'une transaction ou en englober plusieurs.
Le cycle de vie des POJO persistants comprend 4 états distincts faciles à identifier:

- Une **nouvelle entité** ne possède pas d'identité et n'est associée à aucun contexte de persistance.
- Une **entité gérée** possède une identité et est associée à un contexte de persistance.
- Une **entité dissociée** possède une identité mais n'est associée à aucun contexte de persistance.
- Une **entité supprimée** possède une identité, est associée à un contexte de persistance mais est destinée à être retirée de la base de données.

L'interface *EntityManager* définit toutes les méthodes CRUD et de recherche qui font passer un POJO d'un état à un autre. Les attributs persistants peuvent être des types Java primitifs ou leur wrappeur, de type *Date, Collection*, etc. Les associations entre entités sont déclarées à l'aide d'annotations standards : *@OneToOne, @OneToMany, @ManyToOne, @ManyToMany*, chacune pouvant être uni ou bidirectionnelle. Les types génériques Java 5 sont utilisés dans les associations multiples. Voici un exemple d'association multiple:

```
@Entity
public class Project
{
   private Collection<Employee> emp;
   @ManyToMany
   public Collection<Employee>
getEmployees() {return emp;}
   public
void setEmployees(Collection<Employee>
employees) {this.emp=emp;}
   ...
}

@Entity
public class Employee
{
   private Collection<Project> proj;

   @ManyToMany(mappedBy="employees")
   public Collection<Project>
getProjects() {return proj;}
   public void setProjects
(Collection<Project> projects) {this.proj = proj;}
   ...
}
```



Les entités peuvent obéir à des relations d'héritages arbitraires, les relations sont toutes polymorphes. Une entité abstraite peut être persistante, comme il se doit !
Le langage de requête objet est une mise à jour substantielle de l'ancien EJB-QL. Il rajoute de nouvelles opérations telles que JOIN, GROUP BY, la projection HAVING ainsi que des traitements de données en masse. Les requêtes sont statiques ou dynamiques. Elles sont toutes polymorphes. La classe Query fournit une API simple d'exécution des requêtes EJB-QL. Fait remarquable, la notion de gestion optimisée des verrous a été standardisée indépendamment du support de persistance.
Largement inspirée du framework Hibernate, cette API de persistance constitue, pour la première fois dans le monde Java une véritable standardisation du mapping objet-relationnel jusqu'ici confiné à d'obscurs descripteurs de déploiement propriétaires !

Construction simplifiée des WebServices

L'API JAX-WS 2.0 succède à JAX-RPC 1.1. Le modèle de programmation reste essentiellement le même, mais des améliorations ont été apportées en terme d'indépendance du protocole sous-jacent (REST ou SOAP) et d'interopérabilité. Les WebServices bénéficient aussi de l'API JAXB 2.0 pour tout ce qui concerne la conversion de types Java en documents XML. Ceci permet la génération automatique des schémas XML et leur incorporation automatique au document WSDL associé au WebService.
Le mécanisme des annotations vient là encore simplifier considérablement le code des WebServices et de leur client. Pour preuve, voici un WebService et son client dont la simplicité dispense de commentaires :

```
@WebService()
public class Hello
{
   private String message =
new String("Hello, ");
   public void Hello() {}

   @WebMethod()
   public String sayHello(String name)
{return message + name + ".";}
}

public class HelloClient
{
   @WebServiceRef(wsdlLocation=
"http://localhost:8080/helloservice/
hello?wsdl")
      static HelloService service;
   ...

   public void doTest(String[] args)
   {
      try
      {
         Hello port = service.getHelloPort();
         String response = port.sayHello(name);
         ...
      }
      ...
   }
}
```

Enfin, notons que API JAX-WS propose une invocation asynchrone qui dispense le développeur de créer ses propres Threads, ce qui est utile dans les situations où le temps d'invocation est imprévisible.

Autres simplifications

Parmi les autres simplifications bienvenues, en voici deux : le descripteur application.xml n'est plus obligatoire, car le contenu en modules d'une archive EAR est déterminé par introspection. Un répertoire lib à la racine de l'archive est destiné à contenir les librairies partagées entre composants, ce qui dispense l'édition fastidieuse du manifeste de l'application pour la définition de classpath.

L'intégration de technologies existantes

La plate-forme Java EE 5 intègre la spécification JSP 2.1. Celle-ci comprend en particulier la librairie de balises standard JSTL et le modèle de composants web JSF 1.2. La librairie JSTL

existe depuis la spécification JSP 1.2 mais elle fait désormais partie intégrante de la plate-forme, au même titre que le package *java.util* par exemple. En fait la JSTL regroupe 4 librairies, dont la plus importante est la librairie *Core*. Elle propose des balises (*set, out, remove, catch*) permettant d'accéder aux objets dans les différents contextes de la page JSP, des balises de contrôle de flux (*if, choose, forEach, forTokens*) et de gestion des URL (*import, url, redirect*). Pour se référer aux différents objets, les balises JSTL utilisent dans leurs attributs le langage de scripting *EL* (Expression Language).

La spécification JSF 1.2 définit un modèle de composants graphiques web destiné à faciliter la conception d'IHM complexes du point de vue évènementiel. Le framework JSF vise à rapprocher la conception d'IHM web de la conception évènementielle classique (en Swing par exemple). D'une part, il définit une API pour gérer l'état des composants graphiques, la logique de navigation, les validations de surface ainsi que pour réagir aux évènements utilisateurs. D'autre part, JSF définit une librairie de balises pour l'intégration des composants JSF dans une page JSP.

Le cycle de vie d'un composant JSF demeure complexe, il ne comporte pas moins de six phases distinctes : *restauration, extraction des paramètres, validation, mise à jour, gestion des événements et rendu.* On peut dès lors se demander ici si la simplification annoncée est vraiment au rendez-vous. La réponse dépendra certainement de l'existence d'outils visuels, ergonomiques pour le développement de pages JSP avec des balises JSF.

Avant Java EE 5, les balises JSTL et JSF cohabitaient mal en particulier à cause d'incompatibilités entre les versions similaires mais distinctes de *EL* utilisées par les deux librairies. C'est désormais un problème appartenant au passé : il n'existe plus qu'un seul *EL* unifié et enrichi. Ce *Unified EL* permet en particulier de différer l'évaluation des expressions *EL* pour tenir compte du cycle de vie des composants JSF.

La compatibilité avec J2EE 1.4

Naturellement, aucune compatibilité descendante n'est garantie : si un code source utilise les nouvelles API Java EE 5 il ne pourra pas s'exécuter sur une plate-forme J2EE 1.4. En ce qui concerne la compatibilité ascendante celle-ci est en principe assurée dans une grande majorité des cas : tout code existant devrait en principe s'exécuter sur l'implémentation Java EE 5 de référence : Sun Java System Application Server 9. Les quelques exceptions sont dûment répertoriées dans le chapitre *Java EE 5 Compatibility Issues* du *Glassfish Review Draft.* En ce qui concerne les EJB, les EJB 3.0 pourront être des clients des EJB 2.1 et inversement, ce qui devrait notablement faciliter la tâche de migration. La nouvelle version du langage EJB-QL pourra être utilisée dans les méthodes finder() des anciens EJB 2.1.

Nous avons déjà évoqué la compatibilité des *EL* pour la JSTL et JSF qui est au cœur de l'intégration officielle de ces deux technologies dans la nouvelle plate-forme. La présence de la chaîne de caractère #{ utilisée dans le *EL* de JSP 2.1 pour désigner une évaluation différée est interprétée différemment dans un conteneur JSP 2.0. Mis à part ce détail, toute application JSP 2.0 s'exécutera sans problème dans un conteneur JSP 2.1.

Pour se faire une idée

Ce rapide survol vous aura certainement mis l'eau à la bouche. Le meilleur moyen pour vous faire une idée de cette technologie est de jouer avec celle-ci. En attendant la release définitive, Sun propose à cet effet une preview du SDK Java EE 5 qui constitue l'implémentation de référence. Un tutorial Java EE 5 est également disponible sur le site de Sun. Mentionnons encore le projet GlassFish, dont l'objectif est de développer un serveur d'application Open Source compatible Java EE 5. Du côté des environnements de développement, NetBeans 5.5 propose un IDE gratuit avec des assistants pour la construction d'EJB 3.0 et de composants JSF ainsi qu'une autocomplétion de code pour toutes les API Java EE 5. BEA Workshop Studio 3.0 propose une solution commerciale d'outils pour les EJB 3.0 et un outil JSF WYSIWYG.

Conclusion

A notre avis, l'usage généralisé des annotations constitue la contribution majeure à la simplification de la plate-forme Java EE 5. Les technologies EJB 3.0 et JSF quant à elles doivent encore faire leurs preuves. Dans les deux cas leur adoption dépendra pour l'essentiel de la disponibilité d'outils de productivité bien conçus et stables. Java EE 5 une grande simplification ? Certes non ! Mais un premier pas modeste dans la bonne direction.

Pirmin Lemberger

Consultant du Groupe SQLI

FORUM JAVA www.programmez.com

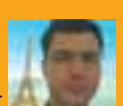
Dialoguez avec les experts Java à partir du 2 Mai

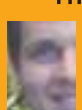
« Le nouveau Java, toutes les questions que vous vous posez »

Avec nos invités experts :

◀ **Hervé Crespel,** Président du Club des Utilisateurs de Java

• **Thibaut RÉGNIER,** Secrétaire Général du Club des Utilisateurs de Java ▶





◀ **Alexis Moussine-Pouchkine,** architecte Java, Sun France

• *Animateur :*

Frédéric Mazué, Programmez ▶

Développez des EJB3 avec EasyBeans

De nos jours, les entreprises cherchent à centraliser leurs systèmes d'informations mais aussi à les rendre intéropérables. Les serveurs d'application répondent à ce besoin. La technologie J2EE est la plus répandue dans ce secteur. Interopérable, standardisée, supportée par la majorité des éditeurs, du marché, elle est particulièrement adaptée aux applications transactionnelles robustes et distribuées.

Les EJB font partie de la spécification J2EE, qui se nomme désormais Java EE. La version 5.0 devrait être finalisée au mois de mai prochain à l'occasion de la conférence JavaOne 2006. L'inconvénient majeur de la version actuelle, version 1.4, concerne la partie persistance et métier avec les EJB 2.1. Les développeurs trouvent cela complexe. Un exemple est la création d'un EJB. Pour créer un objet métier (EJB) rapidement, il faut écrire des fichiers XML (un fichier standard et un fichier spécifique pour chaque serveur d'application) ainsi que plusieurs classes et interfaces Java. Des outils annexes sont apparus afin de simplifier la vie des développeurs J2EE tels que Doclet pour écrire plus rapidement des EJBs ou Hibernate pour simplifier la partie persistance. EJB 3.0 est donc une évolution majeure ayant pour but de simplifier le développement. Cela est possible avec l'adoption de métadonnées : Le code est directement annoté sans passer par un fichier externe (XML). Par exemple, pour définir un EJB, la déclaration de la classe sera préfixée avec l'annotation @Stateless si c'est un EJB sans état. Les descripteurs de déploiement (fichiers XML) deviennent obsolètes, même s'ils sont toujours supportés. Une autre nouveauté des EJB3 concerne la partie persistance avec un modèle proche de celui utilisé par Hibernate. Celui-ci consiste à utiliser des objets standard (POJO: Plain Old Java Object).

Exemple d'utilisation

Nous allons décrire les étapes permettant la construction d'une application métier écrite avec les EJB3. Pour ce faire, le conteneur léger EasyBeans sera utilisé. Ce projet, piloté par Bull et ouvert à tout contributeur, est hébergé par le consortium ObjectWeb. Il servira de conteneur EJB3 au serveur d'application JOnAS, serveur d'application certifié J2EE, lui même hébergé par ObjectWeb. Pour alléger la mise en œuvre de l'exemple, nous utiliserons EasyBeans en standalone à partir des sources du projet. L'outil et un JDK 5.0 sont requis afin de mettre en oeuvre l'exemple.
Les sources d'EasyBeans sont à télécharger sur le site web du projet : "http://www.easybeans.org/download.html"

Installation

Il suffit de décompresser le fichier téléchargé. Par exemple, si le fichier source se nomme *ow_easybeans_src.tgz*, il faut exécuter la commande suivante sous Linux :

jar xf ow_easybeans_src.tgz

Note : Il existe de nombreux outils permettant de décompresser ce type d'archives.

Création d'une application EJB3

Description

L'application qui va être réalisée à titre d'exemple est une bibliothèque. On accède à une interface permettant de lister les livres écrits par différents auteurs. Les deux éléments que sont le livre et l'auteur sont représentés par 2 EJB entités. Ces deux beans ont une relation simpliste : un livre est associé à un seul auteur et pour un auteur donné, il peut y avoir plusieurs livres.

Ecriture des beans entités

Bean Auteur

Une bean entité est une simple classe considérée comme POJO. Cela signifie qu'il faut que la classe comporte un constructeur par défaut. De plus, pour chaque attribut de la classe, un getter et un setter sont positionnés.
Un Bean entité aura comme annotation @Entity. La clef primaire est définie par une annotation @Id. Dans le cas de l'exemple, cette clef doit être autogénérée. Au final, la méthode getId sera la suivante :

```
@Id
    @GeneratedValue(strategy
= GenerationType.AUTO)
    public long getId() {
        return this.id;
    }
```

Le bean Auteur est lié au bean Livre.
Un auteur peut avoir plusieurs livres (relation OneToMany).
La relation est définie également via une annotation :

```
    @OneToMany(mappedBy = "auteur",
fetch = FetchType.EAGER, cascade
= CascadeType.ALL)
    public Collection<Livre> getLivres() {
        return livres;
    }
```

Cette relation est de type One-To-Many. Le lien sera fait avec le bean entité Livre sur l'attribut auteur (propriétaire de la relation).
Le type de fetch est EAGER. Un autre choix peut être Lazy. Dans le cas de l'exemple, les beans entités seront accédés par un client dans une autre JVM (mode détaché). Il ne faut donc pas utiliser le modèle Lazy. Il faut aussi que le bean implémente la classe Serializable car l'accès est fait depuis une autre JVM.
Enfin, le mode cascade est positionné à ALL. Lorsque le bean Auteur sera rendu persistant, les livres attachés à l'auteur seront également persistés.

```
package com.programmez.ejb3;

import java.io.Serializable;
import java.util.ArrayList;
import java.util.Collection;

import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
```

```
import javax.persistence.NamedQuery;
import javax.persistence.OneToMany;

/**
* Definition d'une classe representant
l'auteur d'un livre.
 * @author Florent Benoit
 */
@Entity
@NamedQuery(name =
"tousLesAuteurs", query =
"select o FROM Auteur o")
public class Auteur implements Serializable
{

    /**
     * Attribut pour la serialisation Java.
     */
    private static final long serialVersionUID
= 8279536554568120695L;

    /**
    * Clef primaire (generee automatiquement).
     */
    private long id;

    /**
     * Nom de l'auteur.
     */
    private String nom = null;

    /**
     * Liste de livres ecrits par l'auteur.
     */
    private Collection<Livre> livres;

    /**
     * Constructeur par defaut.
     */
    public Auteur() {
        livres = new ArrayList<Livre>();
    }

    /**
* Constructeur utilise pour initialiser
cet entity bean.
     * @param nom - le nom de l'auteur.
     */
    public Auteur(final String nom) {
        this();
        setNom(nom);
    }

    /**
     * Non utilisation du mode Lazy.
     * @return livres ecrits par cet auteur.
     */
    @OneToMany(mappedBy = "auteur",
fetch = FetchType.EAGER,
cascade = CascadeType.ALL)
    public Collection<Livre> getLivres() {
        return livres;
    }

    /**
     * Ajout d'un livre avec son titre.
     * @param titre - le titre du livre.
     */
    public void ajoutLivre(final String titre) {
        Livre livre = new Livre();
        livre.setTitre(titre);
        livre.setAuteur(this);
        livres.add(livre);
    }

    /**
     * Definition des livres ecrits par cet auteur.
     * @param livres - les livres ecrits.
     */
    public void setLivres(final Collection
<Livre> livres) {
        this.livres = livres;
    }

    /**
     * @return nom de l'auteur.
     */
    public String getNom() {
        return nom;
    }

    /**
     * Definition du nom de l'auteur.
     * @param nom - le nom de l'auteur.
     */
    public void setNom(final String nom) {
        this.nom = nom;
    }

    /**
     * @return un identifiant (auto-incremental)
     */
    @Id
    @GeneratedValue(strategy
= GenerationType.AUTO)
    public long getId() {
        return this.id;
    }

    /**
     * Definition du nouvel identifiant.
     * @param id - nouvel identifiant
     */
    public void setId(final long id) {
        this.id = id;
    }
}
```

Une requête EJB-QL est définie sur la classe du bean. Celle-ci sera utilisée pour obtenir l'ensemble des auteurs.

Bean Livre

Ce bean est également un bean entité. Une relation existe par rapport au bean Auteur, un livre étant lié à un seul auteur : Relation Many-To-One. Cela est défini par l'annotation ManyToOne. L'annotation JoinColumn spécifie ici la colonne de la clef primaire à utiliser lors de l'association entre deux beans entités.

```
/**
    * @return auteur du livre.
    */
    @ManyToOne
    @JoinColumn(name = "Auteur_id")
    public Auteur getAuteur() {
        return auteur;
    }
```

Voici la classe complète :

```
package com.programmez.ejb3;

import java.io.Serializable;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.NamedQuery;

/**
 * Definition d'une classe representant un Liv
re. Un livre comporte un auteur et
 * un titre.
 * @author Florent Benoit
 */
@Entity
@NamedQuery(name = "tousLesLivres",
query = "select o FROM Livre o")
public class Livre implements Serializable {

    /**
```

```
     * Attribut pour la serialisation Java.
     */
    private static final long serialVersionUID
= 7870634228111717036L;

    /**
     * Clef primaire (generee automatiquement).
     */
    private long id;

    /**
     * Auteur du livre.
     */
    private Auteur auteur;

    /**
     * Titre du livre.
     */
    private String titre;

    /**
     * Constructeur par defaut.
     */
    public Livre() {

    }

    /**
     * Constructeur utilise pour initialiser cet
entity bean.
     * @param auteur - l'auteur du livre.
     * @param titre - le titre du livre.
     */
    public Livre(final String titre, final
Auteur auteur) {
        setTitre(titre);
        setAuteur(auteur);
    }

    /**
     * @return auteur du livre.
     */
    @ManyToOne
    @JoinColumn(name = "Auteur_id")
    public Auteur getAuteur() {
        return auteur;
    }

    /**
     * Definition de l'auteur du livre.
     * @param auteur - l'auteur du livre.
     */
    public void setAuteur(final Auteur auteur) {
        this.auteur = auteur;
    }

    /**
     * @return titre du livre.
     */
    public String getTitre() {
        return titre;
    }

    /**
     * Definition du titre du livre.
     * @param titre - le titre du livre.
     */
    public void setTitre(final String titre) {
        this.titre = titre;
    }

    /**
     * @return un identifiant (auto-incremental)
     */
    @Id
    @GeneratedValue(strategy
= GenerationType.AUTO)
    public long getId() {
        return this.id;
    }

    /**
     * Definition du nouvel identifiant.
     * @param id - nouvel identifiant
     */
    public void setId(final long id) {
        this.id = id;
    }
}
```

Fichier de persistance

Il est requis d'avoir un fichier de persistance, META-INF/persistence.xml, permettant de décrire des informations utiles. Par exemple, le datasource à utiliser lors de la persistance en base de données.

```
<persistence
xmlns="http://java.sun.com/xml/ns/
persistence"
            xmlns:xsi="http://www.w3.org/
2001/
XMLSchema-instance"
            xsi:schemaLocation=
"http://java.sun.com/xml/ns/persistence
            http://java.sun.com/xml/ns/
persistence/persistence_1_0.xsd">

<persistence-unit name="entity">
  <provider></provider>
  <jta-data-source>jdbc_1</jta-data-source>
  <properties>
  <property name="hibernate.hbm2
ddl.auto" value="create-drop"/>
  </properties>
</persistence-unit>
</persistence>
```

Bean session sans état accédant aux beans entités

L'accès aux beans entités est réalisé à travers un entity manager, gérant les beans entités. Cet entity manager sera accédé par un bean session sans état.
Ce bean sera utilisé par un client lourd. Ce bean possède 3 méthodes principales définies par une interface :

```
package com.programmez.ejb3;

import java.util.List;

import javax.ejb.Remote;

/**
 * Interface remote du bean jouant le role
du bean "facade".
 * @author Florent Benoit
 */
@Remote
public interface SessionFacadeRemote {

    /**
     * Creation d'auteurs et de livres.
     */
    void init();

    /**
     * @return la liste des auteurs.
     */
    List<Auteur> listeDesAuteurs();
    /**
     * @return la liste des livres
     */
    List<Livre> listeDesLivres();
}
```

Cette interface est utilisée pour accéder au bean depuis un client distant. Elle est donc annotée avec l'annotation @Remote.
Une méthode est nécessaire pour initialiser la base de données avec des livres et auteurs.
Enfin, deux méthodes permettent de récupérer une liste d'auteurs ou de livres. Les objets récupérés sont des objets beans entités.

L'implémentation de cette interface est faite par le bean SessionFacade.
Comme évoqué plus haut, le bean stateless va utiliser un objet entity Manager afin de manipuler les entités. Pour cela, il y a utilisation de la nouvelle fonctionnalité des EJB 3.0 : l'injection de ressource ou "dependency injection".

```
/**
    * Entity manager utilise par ce bean.
    */
   @PersistenceContext
   private EntityManager entityManager = null;
```

L'annotation @PersistenceContext va être analysée par le conteneur et la variable entityManager va être initialisée. Ensuite, il suffit d'appeler des méthodes sur cette variable, elle n'a plus la valeur nulle. Ce n'est plus au développeur de rechercher la variable, il demande au conteneur.
Afin d'ajouter un auteur et ses livres dans une base, il suffit de créer les beans et de les persister au travers de l'entity manager.

```
Auteur balzac = new Auteur
("Honore de Balzac");
Livre pereGloriot = new Livre
("Le Pere Goriot", balzac);
balzac.getLivres().add(pereGloriot);
Livre lesChouans = new Livre
("Les Chouans", balzac);
balzac.getLivres().add(lesChouans);

 // Enregistrement (seulement l'auteur
car il y a l'attribut Cascade positionné à ALL).
 entityManager.persist(balzac);
```

Un bean entité est initialisé comme une classe simple avec l'instruction new(). Cela change des précédentes versions des EJBs (2.1).
On peut ensuite, soit ajouter les livres par une méthode du bean Auteur, soit créér un livre de manière indépendante. Il faut également maintenir la cohérence de la relation, c'est-à-dire ajouter le livre à l'auteur et inversement.
Ensuite, le bean créé doit être rendu persistant dans la base de données. Cela est réalisé avec l'opération persist sur l'objet entitymanager en donnant en paramètre le bean entité balzac (dans l'exemple).
Pour obtenir la liste des auteurs, il y a interrogation de l'objet entitymanager. La requête utilisée dans le cadre de l'exemple est celle définie dans la classe entité avec l'annotation NamedQuery.

Petit rappel, le bean contenait :

```
@Entity
@NamedQuery(name =
"tousLesAuteurs", query =
"select o FROM Auteur o")
public class Auteur implements Serializable {
```

Le bean stateless va donc utiliser le nom de la requête comme argument.

```
 /**
    * @return la liste des auteurs
    */
   @SuppressWarnings("unchecked")
   public List<Auteur> listeDesAuteurs() {
       Query auteurs
= entityManager.createNamedQuery
("tousLesAuteurs");
       return auteurs.getResultList();
   }
```

Cette méthode renvoie la liste des auteurs qui sont stockés dans la base de données.

Code complet de la classe :

```
package com.programmez.ejb3;

import java.util.List;

import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;
import javax.persistence.Query;


/**
 * Bean sans etat assurant le dialogue
cote serveur avec les beans entites.
 * @author Florent Benoit
 */
@Stateless
public class SessionFacade
implements SessionFacadeRemote {

    /**
     * Entity manager utilise par ce bean.
     */
    @PersistenceContext
    private EntityManager entityManager = null;

    /**
     * Creation d'auteurs et de livres.
     */
    public void init() {
        // livres de balzac
        Auteur balzac = new Auteur
("Honore de Balzac");
        Livre pereGloriot = new Livre
("Le Pere Goriot", balzac);
        balzac.getLivres().add(pereGloriot);
        Livre lesChouans = new Livre
("Les Chouans", balzac);
        balzac.getLivres().add(lesChouans);

        // Enregistrement (seulement l'auteur
car il y a l'attribut Cascade positionné à ALL).
        entityManager.persist(balzac);

        // Livres victor hugo
       Auteur hugo = new Auteur("Victor Hugo");
        hugo.ajoutLivre("Les Miserables");
        hugo.ajoutLivre("Notre-Dame de Paris");
        // Enregistrement
        entityManager.persist(hugo);
    }

    /**
     * @return la liste des auteurs
     */
    @SuppressWarnings("unchecked")
    public List<Auteur> listeDesAuteurs() {
        Query auteurs
= entityManager.createNamedQuery
("tousLesAuteurs");
        return auteurs.getResultList();
    }

    /**
     * @return la liste des livres
     */
    @SuppressWarnings("unchecked")
    public List<Livre> listeDesLivres() {
       return entityManager.createNamedQuery
("tousLesLivres").getResultList();
    }

}
```

Florent BENOIT
Architecte et développeur au sein de Bull R&D. Il dirige le projet EasyBeans d'ObjectWeb.

Java 6, le rapprochement de Java et du système d'exploitation

La prochaine version de la plate-forme Java de Sun sera résolument beaucoup plus proche du système d'exploitation. Voyons ensemble deux des fonctionnalités importantes apportées par la prochaine version de JSE : Mustang

Le fauve Java 5 tout juste maîtrisé, voilà que Sun arrive avec sa nouvelle plate-forme Java estampillée en version 6 ! Au programme, pas de grands changements au niveau du langage tel qu'avait pu l'apporter Tiger, mais une grande amélioration de l'intégration entre la plate-forme Java et le système d'exploitation. Cet article a été réalisé avec le JDK 6 Bêta (build 1.6.0-bêta-b59g) disponible à cette adresse : http://java.Sun.com/javase/6/download.jsp

Les splash screen arrivent au grand galop !

Une splash screen ou fenêtre de démarrage, est la fenêtre qui s'ouvre au démarrage d'une application, et qui affiche en général le nom, le logo du logiciel et quelques copyrights. Son rôle principal est de faire patienter l'utilisateur lors du chargement de l'application.

Avec les versions précédentes de Java, nous utilisions AWT ou Swing pour créer notre fenêtre de démarrage. Cependant, le temps entre le lancement du programme et l'affichage de cette fenêtre, qui doit être le plus court possible, était relativement long à cause du chargement de la JVM et d'AWT/Swing. La conséquence en était un affichage relativement tardif de la fenêtre de démarrage, du moins c'était le cas avant Java 6 !

Aujourd'hui, Mustang fournit une solution qui permet à une application d'afficher une fenêtre de démarrage très rapidement, avant même le démarrage de la JVM. Pour permettre cela, l'application appelle une 'pré application' (un launcher) qui se charge d'afficher une image de type JPEG, GIF ou PNG (la gestion de l'animation et de la transparence étant bien évidemment prise en compte) dans une fenêtre sans bords.

La mise en place de cette fonctionnalité est très simple, et il existe deux façons de faire appel à une fenêtre de démarrage ; la première via une option de la ligne de commande, la deuxième en utilisant une option dans le fichier manifest de votre JAR.

Pour les applications en ligne de commande, la méthode consiste à ajouter l'option *-splash:‹chemin du fichier›* à la ligne de commande. Par exemple, la ligne de commande java –splash: *Programmez!84.jpg HelloProgrammez* affichera le résultat suivant :

La seconde méthode, si votre application est packagée dans une archive JAR, passe par l'utilisation de l'option *SplashScreen-Image* qui vous permettra de réaliser la même chose que précédemment. Voici le fichier manifest.mf permettant d'effectuer la même opération qu'en ligne de commande:

```
Manifest-Version: 1.0
Main-Class: HelloProgrammez
SplashScreen-Image: Programmez!84.jpg
```

Listing 1

A noter que l'option de la ligne de commande a priorité sur l'option du fichier manifest.

Personnaliser le splash screen

Afficher une image au démarrage de l'application est intéressant, et la plupart des applications se contentent de ce genre de comportement par défaut. Cependant, il peut être intéressant, voire indispensable, de personnaliser et de contrôler le comportement et l'affichage de cette fenêtre de démarrage. C'est pourquoi, il est possible d'accéder à l'objet *SplashScreen* - qui représente votre fenêtre de démarrage - afin de pouvoir effectuer des opérations directement sur la surface d'affichage. Evidemment, vous ne pourrez accéder à cet objet qu'après ouverture de la fenêtre de démarrage et avant la fermeture de celle-ci.

L'accès à cet objet est basé sur le modèle du singleton, de ce fait, il n'existe qu'une seule et unique instance dans toute l'application. L'accès à cette instance se fait par l'appel à la méthode statique *getSplashScreen()*. Si vous n'avez pas créé de fenêtre de démarrage en utilisant la ligne de commande ou l'option du fichier manifest, cette méthode renverra *null*.

Dans notre exemple, nous allons dessiner une barre de progression sur la fenêtre de démarrage. Pour cela, nous devons récupérer un

objet *Graphics2D* qui nous permettra d'effectuer les opérations de dessin. Il est important de comprendre que la surface de dessin n'est pas celle de l'image, mais une couche située au dessus de l'image. Cette couche possède un canal alpha, ce qui vous permettra de faire quelques effets de transparence dans vos créations artistiques de fenêtres de démarrage. L'appel à la méthode *update()* de l'objet SplashScreen vous permettra de rafraîchir immédiatement le rendu sur la fenêtre de démarrage après manipulation de l'objet Graphics associé à celui-ci.
Voici le code complet dépouillé de toute gestion des exceptions pour plus de clarté pour le lecteur :

```
import java.awt.*;

public class SplashDemo {

        private String[] messages
= {"Initialisation",
"Chargement module", "Lancement"};

        public SplashDemo(){
                SplashScreen
splash = SplashScreen.getSplashScreen();
                Graphics2D g
= (Graphics2D)splash.getGraphics();

                // Simulation d'un
chargement
                for(int i = 0; i <
100; i+=2) {
                        renderSplash
(g, i, messages[i%3]);
                        splash.update();
                        try {

Thread.sleep(100);
                        }

catch(InterruptedException e) {
                        }
                }
                splash.close();
// Fermeture du splash screen
        }

        public void renderSplash
(Graphics2D g, int percentage,
String message){
    g.setComposite(AlphaComposite.Clear);
    g.fillRect(10, 240, 250, 40);
    g.setPaintMode();
    g.setColor(Color.BLUE);
    g.drawString(message +
" en cours ...", 10, 255);
    g.fillRect(10, 265,
(int)(percentage * 2), 20);
        }

        public static void main(String[] args) {
                new SplashDemo();
        }
}
```

Listing 2

Nous commençons par récupérer l'objet *SplashScreen*, puis nous lui demandons son objet *Graphics2D* associé. La boucle qui suit simule une activité de chargement de l'application en appelant tous les dixièmes de seconde notre fonction *renderSplash()*. Cette fonction se chargera tout d'abord d'effacer la surface de dessin pour ensuite afficher un message et la barre de progression. La fenêtre de démarrage disparaît dès que la première fenêtre AWT/Swing s'affiche, ou lorsque vous invoquez la méthode close() de l'objet SplashScreen.

Le support des langages de script

Java 6 implémente désormais la JSR 223 qui introduit un framework standard permettant l'intégration et l'exécution de programmes écrits en langages de script, au sein même de la plate-forme Java. La JSR fournit une interface commune permettant d'intégrer une grande variété de langages de script tels que *JavaScript*, *PHP*, *BeanShell*, *Jython*...
Pour le moment, seul le moteur JavaScript Rhino est incorporé dans la version bêta du JDK (1.6.0-beta-b59g), d'ailleurs pour connaître les moteurs de script supportés par votre runtime, vous pourrez exécuter le listing 3.

```
import java.util.List;
import javax.script.ScriptEngineFactory;
import javax.script.ScriptEngineManager;

public class ScriptEngineTest {

        public static void main(String[] args)
{
                ScriptEngine
Manager seManager =
new ScriptEngineManager();
                List<ScriptEngine
Factory> lsEngineFact
= seManager.getEngineFactories();
                for (ScriptEngine
Factory engineFact : lsEngineFact)
{
                System.out.println("Moteur
'" + engineFact.getEngineName()

+ "' pour langage :
" + engineFact.getLanguageName());
                }
        }
}
```

Listing 3

Le package *javax.script* vous fournira toutes les classes nécessaires pour accéder à ce framework. L'interface *ScriptEngine* est la plus importante, celle-ci définit toutes les méthodes devant être supportées par un moteur de script. L'objet de type *ScriptEngineManager* vous permettra de récupérer le moteur de script que vous désirez.
Dans le code suivant nous récupérons l'implémentation d'un moteur JavaScript :

```
ScriptEngineManager
scriptEngienManager =
new ScriptEngineManager();
ScriptEngine jsEngine = scriptEngienManager.
getEngine
ByName("js");
```

La méthode principale d'un moteur de script est eval(), celle-ci exécute un script contenu dans un String ou via l'utilisation d'un objet Reader. L'invocation de cette méthode vous imposera de gérer l'exception ScriptException. Le code suivant exécute une expression JavaScript qui affiche dans la console de sortie une chaîne :

```
try {
        jsEngine.eval("print
('Hello Programmez!')");
} catch (ScriptException e) {
        e.printStackTrace();
}
```

Exécuter un script, c'est sympathique, mais pouvoir associer des objets Java aux variables de ce langage de script, c'est encore plus intéressant ! Voyons à l'aide du listing 4 un exemple de base d'association objet java/variable de script :

```
Integer a = 20;
```

```
jsEngine.put("a", a);
jsEngine.put("b", 5);
jsEngine.put("val", null);
jsEngine.eval("val = a + b;");
System.out.println
("a + b = " + jsEngine.get("val"));
```

Listing 4

Dans un premier temps, nous déclarons et affectons la valeur 20 à un objet de type *Integer*, puis nous associons cet objet avec le nom de variable de script 'a'. Nous effectuons une opération similaire avec la variable de script 'b' grâce à l'autoboxing de Tiger. Puis nous déclarons une valeur qui nous servira de valeur de retour de l'évaluation de l'addition de 'a' et de 'b'. Celle-ci peut-être null puisque l'objet de retour sera créée par le moteur de script, nous le récupérons d'ailleurs à l'aide de la méthode *get()*, puis nous affichons le résultat de l'évaluation de l'expression :

a + b = 25.0

A noter que l'objet retourné lors d'une addition de deux numériques est de type *Double* et non de type *Integer*, donc faites attention lors des conversions !

Aller plus loin avec le scripting

Ce que nous venons de voir est juste la surface de ce framework de scripting. Les interfaces *Compilable et Invocable* permettent également de faire des choses très intéressantes. Celles-ci offrent respectivement la possibilité de compiler un script pour pouvoir appeler à nouveau celui-ci si nécessaire, et d'appeler une fonction spécifique d'un script préalablement exécuté.

Les spécifications de la JSR 223 décrivent également un framework pour le scripting web articulé autour du package *javax.script.http et des classes HttpScriptContext et HttpScriptServlet*. On peut s'attendre à ce que les futures versions de J2EE intègrent ce Web Scripting Framework en standard.

L'utilisation de langages de script dans des applications n'est pas nouvelle, et se justifie particulièrement si vous voulez fournir un point d'extension pour les développeurs ou les utilisateurs avancés. L'idée étant de permettre l'ajout de fonctionnalités dans une application par l'utilisation de scripts.

Avec l'intégration de cette JSR, Mustang se dote d'un framework puissant pour permettre la création d'applications plus souples et plus flexibles.

Les autres apports de Mustang

Il faudrait encore de nombreuses pages pour traiter de toutes les nouvelles fonctionnalités de Java 6. Par exemple, il faut noter l'évolution de JMX (Java Management Extensions) et de la console Java (JConcole), la disponibilité de JAX-WS 2.0 et JAXB 2.0 pour les Web Services, l'intégration de la JSR 173 avec StAX (Streaming API for XML), le support de l'API Desktop, la gestion de la barre système et des droits sur les fichiers, ...

Au final, Mustang se révèle être une plate-forme plus mature et plus simple que les précédentes versions. Une édition qui apporte aussi son lot de nouveautés et qui semble être une bonne transition entre l'exceptionnelle plate-forme Java 5 et la future édition au nom de code Dolphin.

Liens

JSR 223 : http://www.jcp.org/aboutJava/communityprocess/pr/jsr223/

Site officiel de Java : http://java.Sun.com

Java.net : http://www.java.net/

La javadoc en version bêta se trouve à cette adresse : http://download.java.net/jdk6/docs/api/index.html.

■ Julien Chable
j.chable@proxiad.com
Administrateur Codes-Sources
Blog : http://blogs.developpeur.org/neodante

Migrer vers Java 6

Java 6 n'apportant qu'un lot de nouvelles API, la compatibilité avec l'ensemble des IDE du marché ne devrait pas être un obstacle. D'ailleurs, les environnements de développement Eclipse 3.1 et Netbeans 5.x, pour ne citer que les plus importants du marché, permettent déjà d'utiliser la plate-forme Java 6 de la même manière que la version 5.

Au final, la migration de cette édition ne devrait pas être trop douloureuse pour les entreprises ayant déjà migré vers Java 5. En revanche, pour les autres, la migration d'une version antérieure vers la version 6 aura les mêmes coûts et impacts que le passage vers JSE 5.

Pratique

Java Mustang, une JVM plus proche de votre système d'exploitation

Sun nous concocte son Java 6.0 pour la rentrée prochaine. Découvrons en avant-première deux petites nouveautés sympathiques.

Java 5 à peine apprivoisé, Sun fait évoluer sa JVM au grand galop et nous promet Java 6.0, alias Mustang pour la rentrée. Serons nous bientôt submergés par Java 7, alias Dolphin déjà en préparation ? L'avenir le dira. En attendant, regardons deux nouveautés de Mustang qui témoignent d'une collaboration plus étroite de la JVM avec le système hôte. Vous pouvez télécharger une JVM 6.0 bêta et sa Javadoc à https://mustang.dev.java.net/. Les exemples (sur le Cd-Rom) ont été écrits avec Netbeans après configuration du Java Platform Manager pour cette JVM.

1 De nouvelles méthodes pour les fichiers

La classe java.io.File se voit dotée de méthodes de bas niveaux qui permettent de connaître les capacités d'un disque ou d'une partition, ainsi que l'espace qui y est disponible. Avec un petit bémol, car la documentation précise que les valeurs obtenues doivent être considérées comme un ordre de grandeur et non comme des informations d'une précision absolue. D'autres méthodes permettent de positionner les attributs de lecture/écriture et même de rendre un fichier exécutable. Bien évidemment, ceci n'a de sens que pour les systèmes Unix. Sous Windows, la méthode setExecutable ne fait rien et retourne true systématiquement. Voici un exemple qui illustre tout ce que nous venons de dire.

```
import java.io.*;

public class IODemo {

    public IODemo() {
    }

    public static void main(String[] args) {
        long freespace, totalspace, usablespace;
        String disk = "C:";

        File f = new File(disk);
        freespace = f.getFreeSpace();
        totalspace = f.getTotalSpace();
        usablespace = f.getUsableSpace();

        System.out.printf("Espace libre
sur %s %X\n",
disk, freespace);
        System.out.printf("Espace total
sur %s %X\n",
disk, totalspace);
        System.out.printf("Espace utilisable
sur %s %X\n",
disk, usablespace);

        File texte = new File("C:/texte.txt");
        try {
            if(!texte.exists())
                texte.createNewFile();
            System.out.println
("" + texte.setExecutable(true));
        }
        catch(Exception e) {
            e.printStackTrace();
        }

    }
}
```

2 Accéder à la barre des tâches

Désormais, une application Java pourrait faire acte de présence dans la barre des tâches du système hôte comme le font certaines applications natives. La classe java.awt.SystemTray s'occupe de gérer la question. Le terme System Tray provient de la terminologie KDE et trouve sa correspondance avec la zone de notification de Gnome et la zone d'état de la barre des tâches Windows. Tous ces bureaux sont donc supportés. La classe SystemTray est un singleton créé au démarrage de la JVM. Si la barre des tâches n'est pas disponible, la méthode isSupported doit vous en informer. Questionner cette méthode est incontournable, mais cela ne signifie pas que tout devra aller bien ensuite. Toutefois, si isSupported vous a retourné true, vous pouvez obtenir l'instance par getSystemTray, le point d'accès statique du singleton.

3 Insérer des icônes

SystemTray est un conteneur d'icônes encapsulées par java.awt.TrayIcon. Il est possible d'insérer plusieurs icônes avec SystemTray.add et de les retirer avec SystemTray.remove. Il semble qu'il ne soit pas toujours possible d'ajouter une icône, même si la barre des tâches est signalée au départ comme supportée. La documentation n'est pas très claire pour l'instant sur ce point. Elle nous dit que add peut lever une exception si la barre des tâches n'est pas supportée, ce qui semble curieux car dans ce cas on n'aurait normalement pas accès au singleton. Dans le cas d'un bureau qui ne présenterait pas de barre des tâches, peut être ? Il est par contre certain qu'il n'est pas possible d'insérer deux fois la même icône, sous peine de recevoir une levée d'exception en pleine figure.

4 Personnaliser une icône

Une icône se construit avec au moins une image. A cela, on peut ajouter une bulle d'ai-

DOMAINE - EMAIL - HÉBERGEMENT - CRÉATION DE SITE - E-COMMERCE - RÉFÉRENCEMENT - BACKUP

A partir du 7 avril,
459 millions de personnes
pourront acheter
votre nom de domaine en .eu.

Premier arrivé,
premier servi !

N'attendez pas ! Pour 12 € HT/AN (soit 14,35 € TTC/AN),
protégez votre nom de domaine européen.

AMEN RCS PARIS: B 421 527 797. IN WEB WE TRUST : Nous croyons au web. * Voir conditions sur le site www.amen.fr.
Conditions Générales de Vente sur www.amen.fr. Prix au 01/01/2006, modifiables sans préavis. Photo : Getty : Ryan McVay

Pack Serveur Dédié Baby

Tous les atouts pour vous séduire

- Redhat/Fedora/Windows Server 2003
- Interface d'administration Webmin ou TSE
- AMD Sempron 2200 ou Duron 1600 (ext. à Athlon XP 3000)
- IDE 80 Go (ext. à 2 x 160 Go RAID1)
- RAM 256 Mo (ext. à 2 Go)
- 1 adresse IP fixe (ext. à 4)
- Trafic 1 To (ext. à illimité)

39 € HT/MOIS (à partir de 39 HT/MOIS soit 46,64 € TTC/MOIS pour un engagement semestriel)

50€ OFFERTS

msn adCenter

Trouvez vos meilleurs clients
au meilleur moment, à partir de 0,05 €

Référencez votre site Web en tête du moteur de recherche de MSN, 1er portail Internet français avec 13 millions de visiteurs uniques par mois.

(Médiamétrie//NetRatings - Panel France Applications Internet incluses - Janvier 2006)

50 € DE CLICS OFFERTS*

Avec plus de **200 000 noms de domaine gérés**, AMEN est l'un des principaux fournisseurs européens de services Internet. AMEN est Registrar Officiel auprès de l'EURid (European Registry of Internet Domain Names).

AMEN, Partenaire Officiel du salon WEBMASTERS-EXPO 2006, aura le plaisir de vous accueillir du 3 au 4 mai 2006, Stand 21, à l'Espace Condorcet, La Cité des sciences et de l'industrie, La Villette - Paris.

Compatibles &

0892 55 66 77
(0,34 € TTC/mn depuis la France 9H - 19H)

www.amen.fr

amen
IN WEB WE TRUST

de et un menu surgissant. En ce qui concerne l'image, le lecteur voudra bien adapter le code d'exemple pour que le chargement soit possible. On notera encore l'invocation à la méthode setImageAutoSize(true) sans laquelle nous avons obtenu un résultat complètement aberrant avec notre JVM bêta sous Windows, bien que nous ayons fourni une image 16x16.

Notre icône dotée d'une bulle d'aide de bon conseil.

Notre icône et son menu surgissant à une entrée.

Le texte de la bulle d'aide est défini par une simple chaîne. Le menu est un menu surgissant awt normal. La gestion des événements est tout à fait classique. L'icône peut se voir dotée d'un ActionListener pour réagir au double-clic de souris, ainsi que d'un MouseListener pour réagir au simple clic et aux mouvement du pointeur. En ce qui concerne ces derniers, cela n'a pas fonctionné très bien

Notre icône affiche un message d'information sous la forme d'une bulle d'aide.

avec notre bêta d'essai pour les événements de déplacement du pointeur. Remarquons pour terminer, la méthode displayMessage qui permet d'afficher un message sous la forme d'une bulle d'aide avec titre et bouton de fermeture.

```
// Manipulation de la barre des tâches
avec Java 6.0


public class SystemTrayDemo {

    private
            TrayIcon trayIcon;
    public SystemTrayDemo() {
        if (!SystemTray.isSupported()) {
            System.out.println(
"Votre système ne supporte
pas le SystemTray");
            System.exit(0);
        }
        SystemTray
tray = SystemTray.getSystemTray();
        Image image =
Toolkit. getDefaultToolkit().getImage(
"c:/progicon.gif");
        MouseListener mouseListener =
                new MouseListener() {
            public void mouseClicked
(MouseEvent e) {
                System.out.println
("Evenement Mouse Clicked");
            }
            public void mouseEntered
(MouseEvent e) {
                System.out.println
("Evenement Mouse entered!");
            }
            public void mouseExited
(MouseEvent e) {
                System.out.println
("Evenement Mouse exited!");
            }
            public void mousePressed
(MouseEvent e) {
                System.out.println
("Evenement Mouse pressed!");
            }
            public void mouseReleased
(MouseEvent e) {
                System.out.println
("Evenement Mouse released!");
            }
        };

        ActionListener exitListener =
                new ActionListener() {
            public
void actionPerformed(ActionEvent e) {
                trayIcon.displayMessage
("Programmez!",
"A bientôt...", INFO);
                try {
                    Thread.sleep(2000);
                } catch(Exception ex) {
                    ex.printStackTrace();
                } finally {
                    System.exit(0);
                }
            }
        };

        PopupMenu popup = new PopupMenu();
        MenuItem defaultItem =
new MenuItem("Programmez!");
        defaultItem.addActionListener(exitListener);
        popup.add(defaultItem);
        trayIcon = new TrayIcon(image,
"Tray Demo", popup);
        ActionListener actionListener =
                new ActionListener() {

public void actionPerformed(ActionEvent e) {
                System.out.println
("Action: double clic sur l'icône");
            }
        };

        trayIcon.setImageAutoSize(true);
        trayIcon.addActionListener
(actionListener);
        trayIcon.addMouseListener
(mouseListener);
        trayIcon.setToolTip("Abonnez vous :-)");
        try {
            tray.add(trayIcon);
        } catch (AWTException e) {
            System.err.println
("L'icône n'a pas pu être insérée ");
        }
    }

    public static void main(String[] args) {
    SystemTrayDemo
std = new SystemTrayDemo();
    }
}
```

Frédéric Mazué
fmazue@programmez.com

Abonnez-vous en ligne
www.programmez.com

DOMAINE - EMAIL - HÉBERGEMENT - CRÉATION DE SITE - E-COMMERCE - RÉFÉRENCEMENT - BACKUP

Le jour où vous aurez perdu vos données, que ferez-vous ?

1 entreprise sur 3 dépose le bilan après avoir perdu ses données*.
Amen protège vos données pour le prix d'une corde.

- **Sécurité :** vos données sont sauvegardées sur le réseau IBM infogéré 24x7 avec garantie de disponibilité SLA de 99,5 %
- **Simplicité :** toutes vos données (fichiers, contrats, emails, bases de données...) sont sauvegardées en temps réel et accessibles à tout moment depuis votre PC et/ou une interface web sécurisée
- **Evolutivité :** vous protégez de 1 Go à 50 Go de données
- **Assistance :** vous bénéficiez d'une assistance technique par email et par téléphone

120 € HT/an (à partir de 120 € HT/AN soit 143,52 € TTC/AN)

Avec plus de 100 000 clients, AMEN est l'un des principaux fournisseurs européens de présence Internet.
Notre vocation : innover pour vous offrir des services toujours plus performants.
Si vous croyez au web, vous croirez en nous.

AMEN, Partenaire Officiel du salon WEBMASTERS-EXPO 2006, aura le plaisir de vous accueillir du 3 au 4 mai 2006, Stand 21, à l'Espace Condorcet, La Cité des sciences et de l'industrie, La Villette - Paris.

0892 55 66 77

www.amen.fr

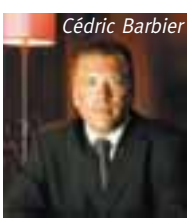
Java, première compétence métier

En 10 ans, Java a su s'étendre et dominer le marché. Certes, la crise est passée par là, mais chiffres et analyses à l'appui, jamais une compétence n'aura été autant recherchée. Enquête.

Sur les termes Java et/ou JEE recherchés dans le descriptif ou l'intitulé du poste dans le secteur informatique, 7 408 offres ont été recensées par l'Apec en 2005, contre 5 058 l'année précédente. Elles représentent respectivement 22% et 20% de la part des fonctions recherchées dans l'informatique, en tête devant toutes les autres. Pierre Lamblin, directeur du département études et recherche de l'Apec, ne semble pas surpris de cette tendance qu'il parvient même à analyser plus en amont : sur 100 offres Java /J2EE, 27 sont dédiées aux ingénieurs en développement, développeurs ou responsables développeurs, 9 pour les ingénieurs ou responsables études et 8 pour chef de projet, directeur de projet ou ingénieur projet.

Pierre Lamblin, photo : Bruno Lévy

Analyse confirmée par Cédric Barbier, fondateur et directeur associé du site d'emploi lesjeudis.com. *« Aujourd'hui, J2EE a dépassé Java. De peu, mais la tendance s'affirme »*. Derrière, C++/C# et .Net se disputent la médaille de bronze. Même constat dans les salons organisés par *lesjeudis* où les recruteurs qui rencontrent les candidats demeurent fidèles à Java, une valeur sûre.



Cédric Barbier

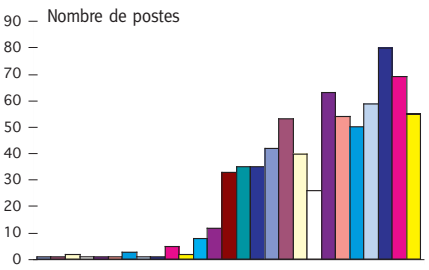
Responsable de la formation clients chez Sun Microsystem, Philippe Argouges va dans le même sens. Il voit passer de nombreux candidats que les entreprises souhaitent formés sur Java. Depuis deux ans, les certifications Java ont été multipliées par cinq *« même si nous partions de zéro ou presque »*. Trois types de certifications sont proposées : pour les programmeurs ; pour les développeurs et pour les architectes.

« Java ? Une solution que l'on peut porter dans tous les environnements »

Pourtant, si Java est recherché, ce n'est pas une compétence à la portée de tous. Elle est enseignée dans la plupart des écoles et universités. La rémunération de ceux qui la maîtrisent peut être dans la fourchette de 30-35 K€, augmentée en fonction de la cote de l'école d'ingénieur. Pour ce diplômé qui aurait des compétences supplémentaires en mathématiques et/ou finances, ce sont de six à dix postes qui s'offrent à lui !

Mais Java seul ne suffit pas. *« Java oui, mais pour quoi faire ? »*, souligne un rien provocateur Pierre François, PDG d'Avolys qui compte 50 centres de formation et propose une cinquantaine de cours différents dans la filière Java/J2EE. *« Tout dépend de la motivation. Je connais quelques autodidactes, mais souvent ceux qui maîtrisent Java ont le portrait robot du passionné tombé dans le développement quand il était petit »*. Avec une tournure d'esprit objet, bien entendu.

Java + J2ee



Une extraction faite sur le site les jeudis montre la bonne santé de Java dans les offres d'emplois. Du 05/2003 au 03/2006.

Cette thèse est confirmée par Eddy Gaciot qui coiffe la triple casquette de professeur au Cnam, à l'Isty et qui dirige GTM-Ingénierie, un cabinet de conseil en ingénierie de la formation, spécialisé en informatique. *« Java est un univers libre, une solution que l'on peut porter dans tous les environnements. Couplé avec UML, il donne des projets intéressants. Mais à une condition, sine qua non : avoir un pré-requis qui s'appelle la méthode »*. Et de développer l'idée pourtant classique que le langage est beaucoup plus facile à appréhender une fois la méthode « digérée ».

Ce ne sont que des personnes à haut potentiel que suit Stéphane Pitout, directeur technique et appui des collaborateurs chez Datavance. Cette SSII de 550 collaborateurs s'est spécialisée dans les prestations intellectuelles de haut niveau pour des clients grands comptes. Java est la composante principale sur ses trois pôles : *« dans la recherche pharmaceutique, Java est le langage des universitaires et de la recherche. Nous nous sommes adaptés à nos chercheurs, il y a une grande effervescence autour de l'outil. Il en est de même dans le domaine de la finance. Avant, les modèles de projections mathématiques étaient en C++. Ils sont aujourd'hui en Java, les trois quarts des mathématiciens l'utilisent. Sa portabilité en fait un outil recherché »*. Quant au troisième secteur dans lequel Datavance est fortement impliquée, il n'échappe pas non plus à Java puisque c'est celui de la téléphonie.

Stéphane Pitout

Alors, quel que soit son Java, léger avec Eclipse ou lourd avec J2EE, avec des préférences pour les Bac+5 mais pourtant quelques récentes timides ouvertures à Bac+2, le candidat doté de cette compétence a toutes les chances de trouver entreprise à son pied et projet à son goût !

■ **Enquête réalisée par Claire Padych**

Les offres en Informatique contenant les termes Java / J2EE - Années 2004 et 2005

	2004	2005
Direction informatique	85	95
Informatique de gestion	3 257	5 07
Informatique industrielle	681	835
Exploitation, maintenance	99	95
Systèmes, réseaux, data	479	562
Autres	457	746
TOTAL INFORMATIQUE	**5 058**	**7 408**
Part sur Ensemble fonction Informatique	20%	22%

* Les termes ont été recherchés dans l'intituléet/ou dans le descriptif du poste. / *Source : offres Apec*

	2004	2005
Ensemble fonction Informatique	25 930	33 481

Genigraph joue la carte Java

Trois questions à Olivier Nicolas, Directeur Produits chez Genigraph (2 agences à Paris et Toulouse)

Quel rapport entre Génitech-Genigraph et Java ?

Genigraph, composante principale du groupe Génitech, est entièrement dédiée à l'Orienté Objet. Elle entretient donc un rapport certain avec Java ! D'une part, dans le service au client avec assistance technique ou forfait. Et d'autre part, avec des solutions métier, notamment e-Citiz, solution de spécification et de développement d'e-services pour l'e-Administration et la dématérialisation en entreprise. Genigraph l'a lancée en novembre 2004 et a reçu le prix de l'innovation des TIC au dernier Salon des Maires et des Collectivités Locales en novembre 2005.

Ce produit logiciel au positionnement métier a été développé en Java, mais il propose une approche MDA, c'est-à-dire que la difficulté technique J2EE (ou Java EE) est masquée. L'application a été pensée au niveau des modèles. Elle a pour but d'apporter des solutions fonctionnelles et techniques aux fournisseurs d'e-Services. e-Citiz est développé en Java et basé sur Eclipse puisque Génitech appartient à la Fondation Eclipse en tant que Plug-In Provider. Nous proposons également AGIR, un progiciel d'aide à la gestion des ressources humaines et matérielles dans le domaine portuaire. D'abord développé en C++, il a évolué en Java.

Pensez-vous que Java séduise les étudiants ?

Java est enseigné dans toutes les écoles. Nous donnons des cours à l'Université Paul Sabatier de Toulouse sur « Java Entreprise » avec un objectif : que les étudiants saisissent comment passer de la théorie à la pratique car c'est le pragmatisme, les réflexes pratiques qui permettent ensuite l'efficacité.

Comment voyez-vous évoluer Java ?

Aujourd'hui, on constate l'émergence des moteurs de règles métiers dans certaines applications Java. Je pense donc que la prochaine étape est l'externalisation des règles métier pour la flexibilité. Nous sommes au début d'une évolution formidable qui permettra d'être de plus en plus fonctionnel et de moins en moins technique ! ■

Hervé Crespel :

« Java a donné ses lettres de noblesse au métier d'architecte logiciel »

Hervé Crespel, président du club des utilisateurs Java depuis 1998, est modeste. « Je ne sais pas bien parler carrière. Mais je peux affirmer que Java a donné ses lettres de noblesse au métier d'architecte logiciel ». Pourtant, c'est lui qui a introduit Java au sein de France Télécom, persuadant le DSI de l'époque de la « révolution » qu'allait engendrer ce langage.

Selon lui, d'ailleurs, il y a un avant et un après Java.

« Avant, le développeur était simplement plus ou moins fort que les autres. Mais aujourd'hui, Java dépasse le langage. Il s'agit d'une architecture d'ensemble. Et la programmation objet est devenue presque banale, disponible sur toutes les plates-formes. Mieux : aujourd'hui l'architecture est devenue portable ».

Autre phénomène lié à Java : une spécialisation de métiers différents. « Java est sur client, sur serveur ou embarqué. Ces métiers ont chacun leurs spécificités. Mais tous ont une syntaxe commune. Java est devenu la compétence de base de l'informaticien. Internet a accentué le phénomène en introduisant une dose complexe dans le développement. Il faut maîtriser une quantité incroyable de technologies ».

Java a changé la façon de penser des développeurs...

Conséquence : Java n'est pas accessible à tous. « C'est vrai que Java a changé la façon de penser des développeurs. Avant, on pouvait faire de la programmation à Bac ou Bac+2. Maintenant, ce sont plutôt des Bac+5. Et en même temps que le métier a été ennobli, sa complexité a augmenté ».

Pourtant, c'est l'aspect Opensource qui est sans doute la plus grande révolution de Java. « Avant, pour monter une activité de développement logiciel, il fallait poser une dizaine de millions d'euros sur la table. Avec Apache, la donne a changé. Sa contribution a été phénoménale puisque toute entreprise pouvait désormais se monter. Et aujourd'hui, Eclipse en est l'apothéose, c'est la meilleure AGI du monde », indique Hervé Crespel.

D'ailleurs, la prochaine étape pourrait être celle de l'atelier universel de tests, digne successeur de l'atelier universel de développement, avec un métier au moins aussi important que celui de développeur : testeur. C'est probablement dans ce sens que les frontières vont évoluer, aidées par l'automatisation des tests. Et tout le monde va travailler encore plus vite... Hervé Crespel, arrête ses prospectives et conclut : « Finalement, Java est une philosophie. Son business model permet de remplacer le prix des licences par celui du service. C'est aussi celui du développement durable ! »

Regards croisés
Java - .Net : le combat des chefs est engagé !

Plutôt Java ou plutôt .Net ? Le schisme est tellement fort entre l'Opensource et la solution propriétaire que tous les yeux sont braqués sur les évolutions des deux technologies et sur la bataille qui s'ouvre entre Eclipse et Visual Studio 2005 !

Pour les offres d'emploi, sur le site lesjeudi.com, on constate une multiplication par presque 3 entre décembre 2005 et février 2006 sur la compétence .Net... et par plus de 38 si l'on remonte à décembre 2004.
Pour l'instant toutefois, et en dépit de cet engouement soudain pour Microsoft revenu à la mode dans les projets, Java garde une réelle longueur d'avance. En « off », des spécialistes du métier indiquent que 23% des projets seraient aujourd'hui sous Java contre 21% en .Net. Il y a 9 mois, la tendance était de 15% en .Net contre 25% en Java. Et en « in », il y a les chiffres. Ceux de l'Apec, par exemple : 22% de compétences Java ou JEE requises en 2005 sur les postes décrits dans le secteur informatique contre 20% l'année précédente. Sur la même recherche, .Net est passé de 9% à 11%. Pourtant, une tension du marché commencerait à se faire sentir sur .Net. La tendance n'est pas étonnante, répond Pierre François, PDG d'Avolys. « La montée de .Net est plus importante que celle de Java. Il y a un an, c'était l'inverse. La plupart des universités ont fait le choix d'enseigner Java. Il est donc logique que la pénurie soit plus forte sur Microsoft lorsque les projets repartent après quelques années d'atonie, puisque clairement le marché se divise entre les deux ! ». Alors, dans les centres de formation répartis sur le territoire, les entreprises demandent de plus en plus souvent que des personnes compétentes en Java JEE soient formées en .Net pour pallier le déficit qui semble de plus en plus visible.

JAVA / J2EE	sur 100 offres
Ing développement, développeur, resp développement	27
Ingénieur études, responsable études	9
Chef de projet, directeur de projet, ingénieur projet	8
Architecte, ingénieur architecte	7
Analyste, analyste programmeur	6
Ing études et développement, resp études et dévelop.	6
Concepteur, ingénieur concepteur	5
Consultant	5
source : offres Apec	

.NET	sur 100 offres
Ing développement, développeur, resp développement	25
Chef de projet, directeur de projet, ingénieur projet	11
Ingénieur études, responsable études	8
Consultant	8
Analyste, analyste programmeur	6
Ing études et développement, resp études et dévelop.	6
Architecte, ingénieur architecte	5
Concepteur, ingénieur concepteur	3
source : offres Apec	

Répartition des emplois pour les compétences en Java/J2EE et .Net

> *« Maîtriser .Net et Java dans un même cerveau peut poser quelques problèmes... »*

« Les deux technologies sont là et elles peuvent cohabiter d'autant plus avec les Webservices », souligne de son côté Hervé Crespel, président du Club Java. « .Net est un monde plus modulaire que communicant. Il faut bien admettre que maîtriser .Net et Java dans un même cerveau peut poser quelques problèmes, même si les deux sont très proches et que les spécialistes français en .Net ont appris Java lors de leurs études ! ». Les recruteurs ne s'y trompent pas : selon l'Apec, les offres exigeant des compétences sur les deux technologies sont seulement de 5% en 2005 contre 3% en 2004. Que faire pour éviter l'inévitable schizophrénie inhérente à ceux qui voudraient concilier le monde de l'Open Source avec celui de la solution propriétaire et ne pas entrer dans la polémique qui sépare le monde en deux blocs ? Apprendre le consensus est une piste. A l'instar de Pierre Lamblin directeur du département études et recherche de l'Apec, qui estime « que les deux compétences sont recherchées sur le marché, reflet des besoins différents des entreprises ».

Regarder ailleurs en est une autre. « Peu d'offres sur le secteur PHP », indique pourtant Cédric Barbier de *lesjeudis.com*. Sentiment confirmé par Pierre François d'Avolys et Eddy Gaciot de GTM -Ingénierie. Mais largement nuancé par Stéphane Pitout, directeur technique de Datavance. « PHP est un outsider que je verrais bien revenir. De nombreux sites sont développés en PHP, les projets sont plus discrets mais ils existent. En plus, le coût est peu élevé et l'outil solide... » Et de conclure, malicieusement : « J'attends aussi la riposte d'Apple qui s'est rapproché de Sony Ericsson avec le Pacte Orange... »

■ **Claire Padych**

TOUT NOUVEAU TOUT BEAU

Programmez.com change !

Photo : © Larry Williams/zefa/Corbis

La V2.0 offre le premier portail de services du développeur.

Les Blogs des développeurs

Vous avez la parole !
On n'attend que vous !

Le Who's Who

Est-ce que vous y êtes ?
Avez-vous complété votre profil ?

Logiciels

Le guide des logiciels et des outils de développement. Téléchargement.

Solutions

Les cas d'entreprises, classés par métier et par type d'applications.

Magazine

Consultation des numéros, sources, liens, et interactivité avec les rédacteurs.

Forums

Echangez vos expériences, donnez votre opinion, et discutez avec les auteurs de Programmez!

WWW.PROGRAMMEZ.COM

Le contrôle qualité des applications Java

Développer une application Java selon les règles de l'art est beaucoup moins aisé qu'il n'y paraît. Les projets, du fait de la complexité de la norme J2EE, sont aujourd'hui dans bien des cas, très volumineux et compliqués. Pour faire face à ces projets, les équipes de développement mises en place sont nécessairement conséquentes et bien souvent hétérogènes. Ceci n'est pas un problème en soi, sauf, bien évidemment, quand les principes et les règles de développement Java sont méconnues et mal implémentées par les développeurs.

Dans ce cas, les projets ont tendance à devenir moins facilement maintenables et évolutifs durant leur cycle de vie. La conséquence immédiate est que les développeurs deviennent difficilement interchangeables, parce qu'ils sont les seuls à maîtriser les développements. La situation peut être encore plus difficile sur des projets ayant accumulé beaucoup de retard. En effet, bien souvent, ce retard ne peut être rattrapé qu'en développant vite et mal. Le résultat prévisible de cette situation est une application instable et difficilement maintenable, parce que non développée selon les règles de l'art.

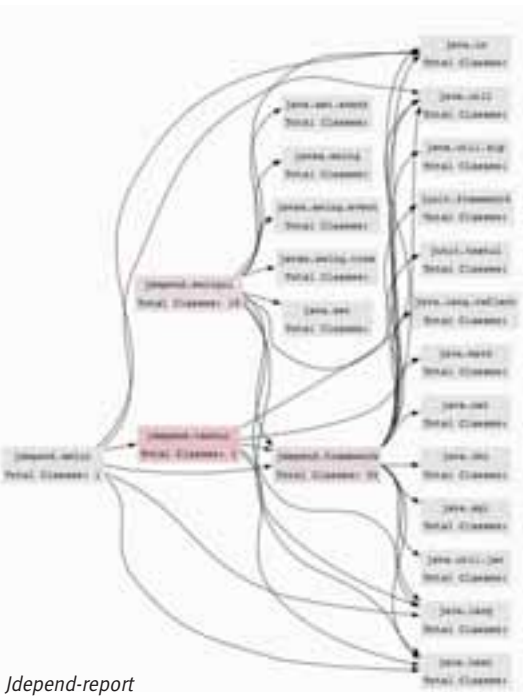
Face à de telles situations, des solutions existent. Tout d'abord, il s'agit de s'assurer des compétences des équipes de développement. Si cela s'avère nécessaire, proposer aux développeurs juniors d'être accompagnés les premiers temps par des développeurs seniors qui les initieront aux bonnes pratiques de développement. Mais enfin et surtout, il faut mettre en œuvre une véritable politique de contrôle qualité de l'application. Malheureusement, ceci est très rarement fait. Cela vient de ce qu'une telle politique est parfois jugée inutile, parfois intrusive mais dans tous les cas, coûteuse et sans perspective de retour sur investissement. C'est seulement dans les moments de crise d'un projet que le contrôle qualité d'une application est évoqué comme solution à tous les maux. Mais pour être efficace, il doit être décidé en amont du projet et implémenté durant tout son cycle de vie. Pour contrôler au mieux le code Java d'une application, il est nécessaire de savoir ce qu'il faut contrôler et comment procéder. Il existe deux types de contrôles. Le premier est le contrôle statique. Son objectif est d'auditer le code source ou compiler sans l'exécuter. Le second est le contrôle dynamique également nommé profiling. Son objectif est d'auditer l'application pendant son exécution.

Le contrôle statique

Le contrôle statique permet de valider le respect des règles de développement Java. Certaines de ces règles sont fondamentales pour permettre une bonne maintenabilité et une bonne évolutivité du code. Ainsi, on s'efforcera de contrôler la complexité cyclomatique des méthodes d'une classe Java. Celle-ci mesure le nombre de branchements conditionnels et d'itérations présentes dans une méthode Java. Une complexité cyclomatique élevée traduit donc un code complexe à maintenir et complexe à tester dans son intégralité. On doit également contrôler l'indépendance des composants d'une application les uns par rapport aux autres. Les composants métiers ne doivent pas, par exemple, êtres dépendants des composants techniques. Une autre règle, plus importante qu'il n'y paraît, est d'avoir un code suffisamment commenté et documenté. Cela est primordial. En effet, un code mal commenté ou mal documenté est difficilement exploitable, même pour son auteur, pour peu qu'il ne soit pas intervenu sur le code depuis plusieurs semaines.

Il existe des outils permettant d'effectuer un contrôle statique du code Java. La communauté open source est très active sur le sujet. Elle propose un grand nombre de produits. Le problème est qu'ils ne sont pas tous convaincants, loin de là. Cependant, trois outils combinés entre eux permettent de couvrir l'ensemble des besoins du contrôle statique de code. Il s'agit de CheckStyle, Metrics et JDepend. CheckStyle est un outil open source proposé sous licence Lesser GPL. Il est téléchargeable depuis l'adresse http://checkstyle.sourceforge.net. Cet outil propose de puissantes fonctionnalités permettant d'appliquer des contrôles sur le respect des règles de développement. Pour définir les contrôles réalisés lors de son exécution, CheckStyle utilise une configuration définie dans un fichier XML qui précise les règles à vérifier et le seuil d'alerte associé à chaque règle. CheckStyle existe notamment sous la forme d'un plug-in compatible avec la plus part des environnements de développement. Ce plug-in permet aux développeurs de contrôler la complexité de leurs développements en temps réel et donc de corriger les éventuelles déviances. Metrics est un plug-in développé par la société Team In A Box Ltd. Il est compatible avec la plateforme de développement Eclipse. Il est proposé sous licence CPL. Il est téléchargeable depuis l'adresse http://www.teaminabox.co.uk/downloads/metrics/.

Ce plug-in permet aux développeurs de contrôler la complexité de leurs développements en temps réel à la manière de CheckStyle. Mais sa force réside dans le fait qu'il permet d'exporter sous forme de résultats consolidés les métriques du projet au format HTML. Le rapport généré permet donc au responsable du projet de contrôler à tout moment l'état global de son projet. JDepend enfin, est un outil open source



Jdepend-report

développé par la société ClarkWare Consulting, Inc. Il est téléchargeable depuis l'adresse http://www.clarkware.com/software/JDepend.html. Cet outil existe, entre autres, sous la forme d'une application graphique Java Swing. JDepend est très utile parce qu'il permet de diagnostiquer des problèmes de conception. En effet, l'outil fournit un arbre représentant les dépendances entres les packages d'une application. Ceci permet de s'assurer de l'indépendance des packages métiers vis-à-vis des packages techniques. De plus, un certain nombre d'informations supplémentaires fournies par l'outil permettent de mesurer le niveau d'instabilité d'un package ou son niveau d'abstraction. Ces informations permettent d'améliorer la maintenabilité du code et donc de faciliter son évolution.

Le profiling

Le profiling permet d'analyser la consommation des ressources système. Il permet également de détecter les éventuels goulots d'étranglement et certains bugs liés aux applications multitâches comme les deadlocks. Java offre une facilité aux développeurs : la libération des objets instanciés n'est pas de leur ressort mais de ce que l'on appelle le garbage collector. Les développeurs ne se sentent donc plus concernés par la gestion des objets en mémoire. Le problème est que lorsque le garbage collector supprime les objets devenus inutiles, les applications sont très fortement ralenties. L'idée est donc de contrôler la manière dont les objets sont créés dans le code Java afin de s'assurer que le garbage collector est sollicité au mieux. Le profiling permet également de s'assurer que des ressources allouées sont bien libérées. Cela est extrêmement important, notamment lorsque les ressources en question sont des connexions vers des bases de données. Les outils de profiling vont donc analyser la JVM sur laquelle une application s'exécute et en extraire des informations sur la consommation des ressources. Ces informations sont ensuite utilisées pour analyse *(Figure 1)*.

La communauté open source propose un grand nombre d'outils dédiés au profiling. Cependant, dans le cadre de projets d'applications d'entreprises, ils s'avèrent, dans la plus part des cas, insuffisants voire inutilisables. En effet, ces outils sont rarement compatibles avec les principaux serveurs d'applications du marché. Quand cela est le cas, ils sont complexes à configurer ou difficiles à mettre en œuvre, certains de ces outils nécessitant même l'ajout d'instructions

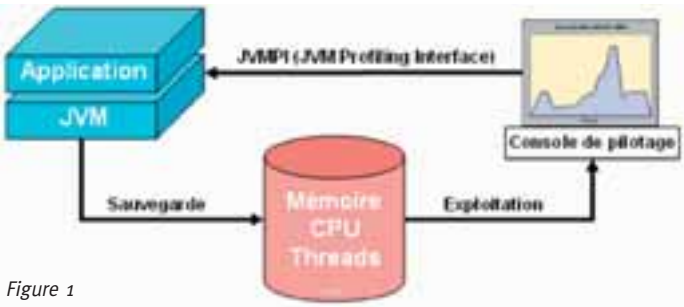


Figure 1

spécifiques dans le code source. Il existe cependant certains outils commerciaux de bonne facture, notamment JProfiler de la société ej-technologies et JProbe de la société Quest software. Tout d'abord, ces deux outils sont compatibles avec la plus part des JVM et serveurs d'applications du marché. De plus, leur mise en œuvre ne nécessite pas de modification du code applicatif. Parmi les fonctionnalités proposées par ces outils, on trouve la possibilité d'analyser la mémoire utilisée par une application. L'objectif est d'en détecter les surconsommations ou les consommations non maîtrisées pouvant donner lieu à des ralentissements de l'application. Ils permettent également d'analyser la consommation CPU. L'objectif est de déterminer le temps d'exécution de chaque composant de l'application. Cela permet notamment de détecter les goulots d'étranglement et donc d'optimiser le code. Il est important de prendre conscience que ce dernier point est primordial. En effet, la diminution des temps de réponse est très appréciée des utilisateurs finaux d'une application. Ces outils permettent également de suivre l'activité des threads d'une application. Cette fonctionnalité permet de détecter les problèmes d'accès concurrents aux ressources et plus particulièrement les deadlocks. Le deadlock surgit lorsque deux threads se bloquent mutuellement parce que chacun tente d'accéder à une ressource réservée par l'autre qui ne peut la libérer. En dehors du profiling, l'origine de ce type de problème est très difficile à déterminer et génère souvent des bugs difficiles à corriger. La mise en œuvre d'une politique de contrôle qualité, nous l'avons vu, permet d'améliorer les applications produites. Cependant, pour être efficace, elle doit implémenter le processus illustré par la figure 2.

Grâce aux outils dédiés, la tâche des équipes de développement est facilitée. Le retour sur investissement est considérable pour plusieurs raisons. D'abord parce que le contrôle qualité permet de mieux maîtriser les développements. Il permet aux développeurs d'acquérir de bonnes pratiques et les sensibilise à l'optimisation. Ensuite, parce que le coût financier est relativement faible. En effet, il est possible de mettre en place un audit statique du code, à moindre coût. Les outils de profiling sont certes payants, mais leur mise en œuvre sur plusieurs projets, les rend vite rentables. Enfin et surtout, les applications produites sont plus stables, plus rapides et cohabitent mieux entre elles parce que auditées et profilées.

Ali Meraoumia
SQLI

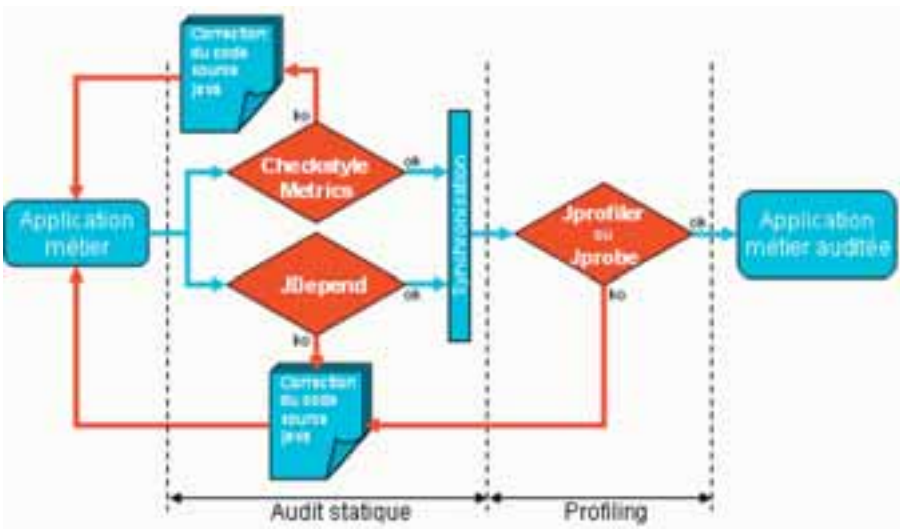


Figure 2

Comment développer une solution RFID ?

Mener à bien le développement d'une solution RFID peut s'avérer relativement complexe sans une approche professionnelle et structurée. En effet, contrairement aux projets informatiques traditionnels, il existe une composante nouvelle : relier le monde physique (les objets sur lesquels les étiquettes RFID sont placés) et les applicatifs de l'entreprise. Et cela nécessite la mise en oeuvre de compétences multiples.

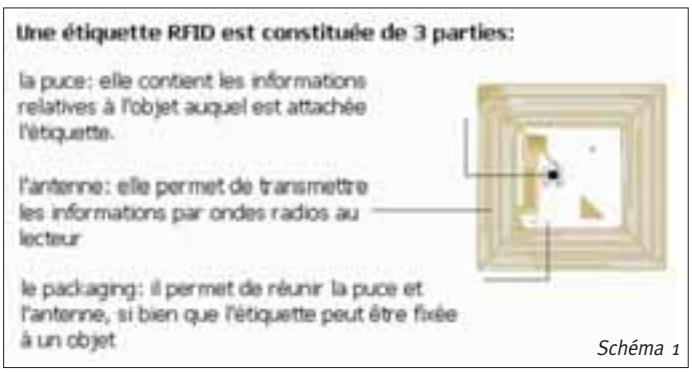


Schéma 1

La RFID (Radio Frequency IDentification) permet la lecture/écriture d'informations sur une étiquette électronique (tag), par exemple un identifiant. On distingue deux types d'étiquettes : actives (munies d'une batterie) ou passives (l'énergie nécessaire au fonctionnement de l'étiquette est reçue des systèmes de lecture par effet d'induction ou par couplage électromagnétique entre le lecteur et l'étiquette) **sch. 1.** Plusieurs gammes de fréquences et technologies existent : on parle d'étiquettes basse fréquence, Haute Fréquence (HF) à 13.56 Mhz, Ultra Haute Fréquence (UHF aux alentours de 900 Mhz). Les distances de lecture et autres caractéristiques varient en fonction des technologies. Une partie du projet de mise en œuvre d'une solution RFID consiste à choisir la meilleure technologie en fonction des contraintes d'usage mais aussi des standards applicables au processus métier envisagé. Ceci nous conduit naturellement à un autre aspect crucial de la réussite de toute initiative autour de la RFID : l'analyse des processus métiers de l'entreprise susceptibles de bénéficier de l'utilisation de la technologie RFID. Une fois les processus métiers identifiés, les impacts du déploiement de technologie RFID évalués et le retour sur investissement chiffré, le ou les projets d'intégration et de déploiement peuvent être lancés.

Schéma 3

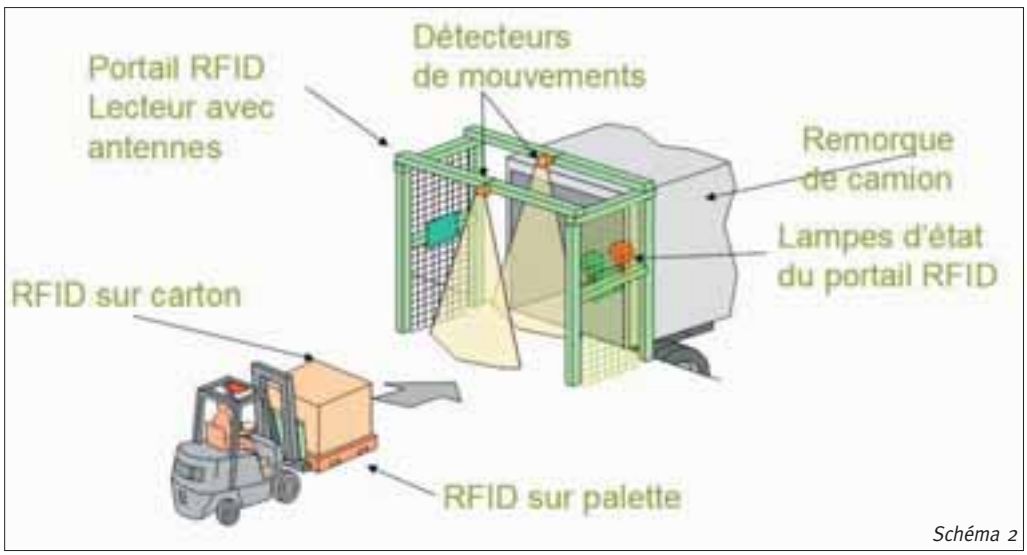


Schéma 2

Exemple de projet

Le suivi de colis ou palettes dans la chaîne logistique, la traçabilité des produits, le pilotage des processus de fabrication, le suivi et l'inventaire de biens dans l'entreprise... font partie des nombreux domaines d'application de la technologie RFID. Prenons pour notre exemple un usage logistique : la préparation et l'expédition de palettes dans un entrepôt **sch.2**.

L'architecture peut se décomposer en trois grands domaines fonctionnels :

1. La gestion et le contrôle du point de lecture (le lecteur RFID et ses antennes, les détecteurs de mouvement indiquant si la palette est chargée ou déchargée du camion, les indicateurs lumineux permettant d'alerter rapidement et simplement le cariste d'une erreur).
2. La transformation des évènements de lecture physique d'étiquettes RFID en évènements métiers significatifs pour le système d'information de l'entreprise et la gestion et configuration des différents points de lecture/écriture RFID.
3. L'intégration des évènements métiers dans les processus de l'entreprise et notamment l'intégration avec la ou les applications intervenant dans ces processus sch.3.

Certains fournisseurs middleware du marché ont développé une offre facilitant la gestion des évènements RFID. Peu sont capables de couvrir ces trois domaines fonctionnels. Les plus avancés ont investi sur une offre embarquée pour gérer les points de lecture préchargés dans les lecteurs RFID : on parle alors de lecteurs intelligents : « Smart Reader ». L'intégration avec le serveur de gestion des évènements RFID peut s'effectuer au travers d'API à base de message XML. Ils permettent ainsi le déploiement d'architecture de type « Event Driven Architecture » et/ou au travers de « Web Service » facilitant le développement de solutions basées sur une architecture de type SOA.

Nous utiliserons une interface de type « Web Service », et plus particulièrement celle spécifiée par le consortium EPCGlobal représenté en France par GS1. Cette interface s'appelle « l'Application Level Event » (ALE) Version 1.0. Elle est disponible à l'adresse suivante : http://www.epcglobalinc.com/standards_technology/EPCglobal_Object_Naming_Service_ONS_v112-2005.pdf

Nous allons d'abord définir le processus métier à réaliser :

- Etape 1 : Préparation de colis pour expédition
- Etape 2 : Mise en palette
- Etape 3 : Vérification automatique des palettes

Schéma 4

- Etape 4 : Suivi et contrôle de l'expédition des palettes
- Etape 5 : Génération automatique de notification d'expédition aux destinataires

A chaque étape les opérations à réaliser seront :

- Etape 1 : Impression et écritures des étiquettes adhésives RFID pour les colis et palettes. Une association entre les identifiants colis et les identifiants palettes existe pour chaque palette dans l'application de l'entreprise.
- Etape 2 : Les étiquettes sont collées sur les colis et la palette. Les colis sont placés sur la palette. Attention : cette opération est très souvent manuelle et peut conduire à des erreurs.
- Etape 3 : Pour une palette donnée, lecture des étiquettes RFID placées sur les colis et sur la palette. Vérification automatique de conformité par rapport aux informations stockées en base. Un indicateur sonore et lumineux indiquera au technicien si la palette est correcte ou pas.
- Etape 4 : Pour le chargement dans un camion donné, les étiquettes RFID des palettes seront lues et une vérification sera effectuée en « temps réel » pour s'assurer que la palette est bien chargée dans le bon camion. Un indicateur lumineux indiquera au technicien si la palette est bien prévue pour le camion en cours de chargement ou pas.
- Etape 5 : En fin de chargement une notification d'expédition sera émise vers le ou les destinataires des palettes ou colis.

Considérons désormais que l'entrepôt à équiper disposera de deux portes de chargement / déchargement et d'une station de vérification automatique de palette sch.4.

Trois points de lecture seront configurés dans le serveur d'évènements RFID avec notamment les informations sur les types de lecteurs utilisés, le ou les contrôleur(s) industriel(s) en charge de leur gestion, les éléments matériels externes connectés au(x) contrôleur(s).

Dans certain cas, la logique de contrôle du point de lecture sera fournie par le middleware. Dans d'autres cas il pourra être nécessaire de l'enrichir dans le cadre du projet. Dans le même temps, des imprimantes d'étiquettes RFID pourront être connectées au contrôleur de point de lecture pour l'impression des étiquettes.

Schéma 6

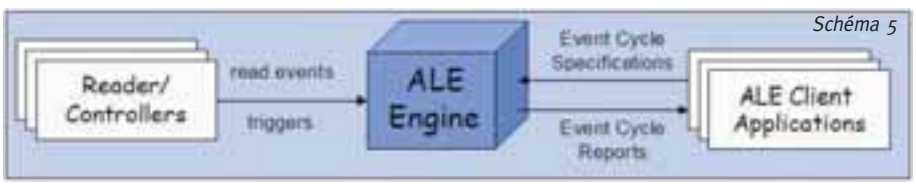


Schéma 5

Application Level Event

La spécification Application Level Event (ALE) 1.0 définit la notion d'API de haut niveau à destination des applications qui n'ont pas

vocation à interagir avec la masse d'évènements générée par les systèmes tels que les lecteurs RFID sch.5.

L'ALE définit une terminologie visant à isoler les caractéristiques des systèmes de capture d'identifiants (aussi appelés EPC : Electronic Product Code), comme les lecteurs RFID, des besoins des applications.
On parlera ainsi de :
• Cycle de lecture (Read cycle): l'unité d'interaction avec un lecteur
• Lecteur logique (Logic reader): un ou plusieurs lecteurs physiques
• Cycle d'évènement (Event cycle) : l'unité d'interaction entre une application cliente et l'interface ALE
• Rapport (report) : il s'agit des données fournies par l'interface ALE à l'application cliente à l'issue d'un cycle d'évènement.

Le schéma 6 tente de résumer ces notions. Des identifiants (EPC1, 2, 3, 4, 5) sont lus par un ou des lecteurs et / ou des rapports sont fournis à trois applications clientes différentes (Client 1, 2 et 3) selon les cycles de chaque application.

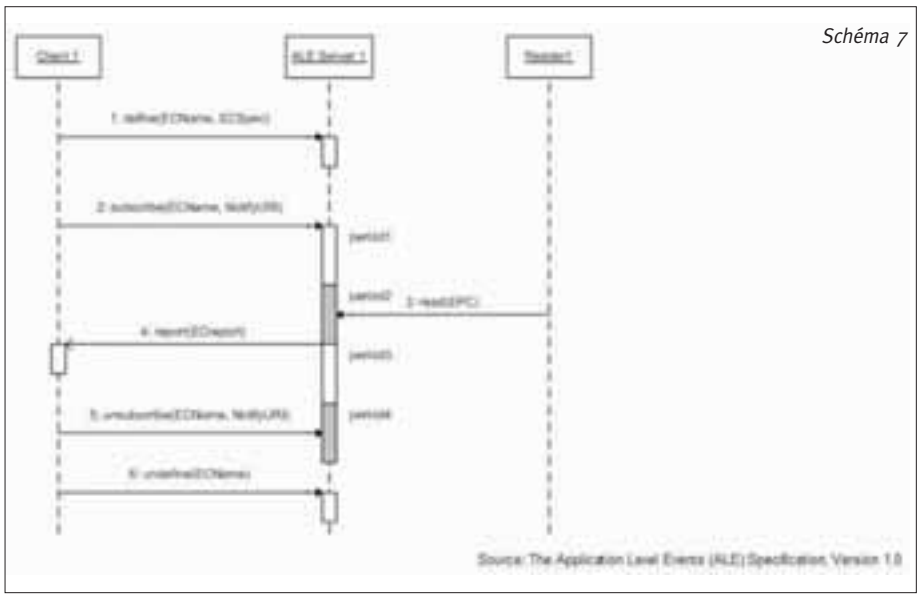


Schéma 7

L'utilisation de l'interface par une application cliente peut se résumer en 6 étapes :
1. L'application spécifie (create) un « event cycle »
2. L'application envoie (define) la spécification de « l'event cycle » à l'ALE
3. L'application s'abonne (subscribe) à « l'event cycle »
4. L'application reçoit des rapports réguliers de l'ALE
5. L'application se désabonne (unsubscribe) à « l'event cycle »
6. L'application détruit (undefine) la spécification de « l'event cycle » sch.7

Un mode simplifié (immediate) permet d'utiliser un mode d'interaction synchrone avec l'interface. Dans ce cas, l'application définit l'event cycle et reçoit un rapport en un appel synchrone à l'ALE. Un autre mode simplifié permet de « poller » l'interface régulièrement.

Un « event cycle » se caractérise par :
• Une ou des sources de données :
i. Nom de lecteur(s) logique(s)
• Des paramètres d'agrégation de données :
i. La période de collecte : La durée
ii. Les critères de sélection des étiquettes : étiquettes palettes, étiquette cartons, étiquette produit.
• Des formats de rapport
i. Toutes les étiquettes lues dans ce cycle
ii. Toutes les nouvelles étiquettes lues dans ce cycle par rapport au cycle précédent
Il faut considérer un autre aspect important dans tout projet RFID : les données inscrites dans la mémoire de l'étiquette RFID. Il est possible d'inscrire un identifiant ou plusieurs kilobit d'information. Le format des données pourra être propriétaire ou standard. Dans cet exemple, le format des données s'appuie sur le standard de l'EPCGlobal EPC Generation 1 Tag Data Standards disponible à l'adresse suivante : http://www.epcglobalinc.com/standards_technology/Ratified_Spec_March_8_2006.pdf

Programmation

Nous utilisons la plate-forme J2EE. Il est bien sûr possible d'utiliser, notamment en architecture SOA, des middlewares d'intégration. Ils permettent l'exécution de processus décrits sous forme de documents standard BPEL (Business Process Execution Language). Nous nous contenterons ici de développer du code Java en utilisant un environnement de développement basé sur Eclipse. Ce code pourra être déployé sur le même serveur que le middleware de gestion des évènements RFID.
Pour simplifier l'exemple, nous ne traiterons dans la suite de cet article que les étapes 3 et 4. Nous considérons aussi que les palettes seront identifiées par des tags dont la mémoire stockera un Serial Shipping Container Code (SSCC) sur 96 bits et les colis de produits seront identifiés par des tags dont la mémoire stockera un Global Trade Identification Number (SGTIN) sur 96 bits.

Étape 3 : Pour une palette donnée, lecture des étiquettes RFID placées sur les colis et sur la palette. Vérification automatique de conformité par rapport aux informations stockées en base. Un indicateur sonore et lumineux indiquera au technicien si la palette est correcte ou pas. Le processus pourra être le suivant :
1. Le technicien passe une palette à l'aide d'un transpalette sous le portique de vérification
2. Le détecteur de mouvement connecté au contrôleur de point de lecture **« Verif 1 »** démarre le lecteur correspondant.
3. Les identifiants de la palette et des colis sont lus, transmis au contrôleur de point de lecture qui filtre les doublons. Plusieurs choix d'implémentation sont alors possibles : soit les identifiants lus peuvent être transmis « au fil de l'eau » au serveur d'évènements RFID, soit ils peuvent être agrégés par le contrôleur pour être transmis « d'un bloc » ; par exemple, lorsqu'un autre détecteur constate la sortie du portique de vérification.
4. Le composant logiciel en charge de la vérification reçoit le rapport ALE contenant tous les identifiants. Il interroge la base applicative et vérifie si la palette est connue et si tous les colis associés à cette palette sont bien présents.
5. Il informe le serveur d'évènements RFID qui actionnera l'indicateur lumineux pour informer le technicien du résultat de la vérification. Par exemple : Vert (Palette conforme), Rouge (Palette inconnue), Orange (Erreur de colis).
Le détecteur de sortie arrête le lecteur RFID.

Il est à noter que les étapes 2, 3, 5, 6 doivent être si possible gérées par le middleware RFID pour les interactions avec l'applicatif et apporter une plus grande valeur au concepteur de la solution.

• L'event cycle utilisé pourra être :
Sources de données :
■ Nom de lecteur(s) logique(s) : **Verif 1**
Des paramètres d'agrégation de données :
■ La période de collecte : **10 secondes**
■ Les critères de sélection des étiquettes : **étiquettes palettes, étiquette cartons**
Des formats de rapport
■ **Toutes les nouvelles étiquettes lues dans ce cycle par rapport au cycle précédent**

Pour l'étape 3 nous choisissons une souscription permettant de recevoir les évènements par appel d'une servlet :

```
package test;

import java.util.PropertyResourceBundle;

import ale.helper.ALETesterECReportSpec;
import ale.helper.ALETesterECSpec;

import com.ibm.rfid.epcglobal.ale.soap.
ALEServicePortTypeProxy;
import com.ibm.rfid.epcglobal.ale.soap.
Define;
import com.ibm.rfid.epcglobal.ale.soap.
Subscribe;
import com.ibm.rfid.epcglobal.ale.types.
ECReportSpec;
import com.ibm.rfid.epcglobal.ale.types.
ECReportSpecs;
import com.ibm.rfid.epcglobal.ale.types.
ECSpec;

public class SubscribeWithServlet {

        private static String
PROPERTIES_FILE = "ALEAPITestEtape3";

        public static void main(String[] args) {
PropertyResourceBundle
resProp = (PropertyResource
Bundle) PropertyResourceBundle.getBundle
(PROPERTIES_FILE);
                String ALE_URL
= resProp.getString("ALE_URL");
                String notificationURI
= resProp.getString("notificationURI");
                String[] readers
= new String[] { resProp.getString
("locationId") };
                String[] includePattern
= new
String[] {resProp.getString("includePattern")};
                String duration
= resProp.getString("duration");

                // Get the ALE
engine proxy
                ALEServicePortType
Proxy proxy = new ALEServicePortTypeProxy();
                proxy.setEndpoint
(ALE_URL);

                // Create an ECSpec
for the specified readers
                ECSpec ecSpec
= ALETesterECSpec.createECSpec(readers,
duration);

                // Get an ECReportSpec
        ECReportSpec additionsReportSpec
= ALETesterECReportSpec.createECReportSpec
("ADDITIONS",
"AdditionsReport", includePattern);
                ECReport
Spec deletionsReportSpec
= ALETesterECReportSpec.createECReportSpec
("DELETIONS",
"DeletionsReport", includePattern);
                ECReport
Specs reportSpecs = new ECReportSpecs();
                reportSpecs.setReportSpec
(new
ECReportSpec[] { additionsReport
Spec, deletionsReportSpec });

                // Add
the ECReportSpecs to the ECSpec
                ecSpec.setReportSpecs
(reportSpecs);
                ecSpec.setIncludeSpecIn
Reports(true);

                // Define and
subscribe the spec to the ALE engine
                Define define =
new Define();
                Subscribe subscribe
= new
Subscribe();
                define.setSpec(ecSpec);
                define.setSpecName
("AdditionSetTestSpec");
                subscribe.setSpecName
("AdditionSetTestSpec");
                subscribe.setNotification
URI
(notificationURI);
                try {
                        proxy.define
(define);
                        proxy.
subscribe(subscribe);
                } catch (Exception e)
{
                        System.out.
println("Exception: " + e.getMessage());
                }
        }
}
```

avec le fichier de propriété ALEAPITestEtape3 suivant :

```
ALE_URL=http://9.212.11.71:
9080/CandidateAleSoapService/services/
ALEServicePort
notificationURI=http://9.212.1.40:9080/
ALECallbackServlet/ALECallbackServlet
locationId=Verif1
duration=10000
includePattern=urn:epc:pat:*.*.*.*
specName=CurrentSetSpec
reportName=CurrentTagsReport
reportSet=CURRENT
```

Etape 4 :
Pour le chargement dans un camion donné, les étiquettes RFID des palettes seront lues et une vérification sera effectuée en « temps réel » pour s'assurer que la palette est bien chargée dans le bon camion. Un indicateur lumineux indiquera au technicien si la palette est bien prévue pour le camion en cours de chargement ou pas.

• L'event cycle utilisé pourra être :

Sources de données :
■ Nom de lecteur(s) logique(s) : **Porte 1 ou 2**
Des paramètres d'agrégation de données :
■ La période de collecte : **5 secondes**
■ Les critères de sélection des étiquettes : **étiquettes palettes**

Des formats de rapport
■ **Toutes les nouvelles étiquettes lues dans ce cycle par rapport au cycle précédent**

Pour l'étape 4 nous choisissons une souscription permettant de recevoir les évènements dans une queue de messages JMS :

```
package test;

import java.util.PropertyResourceBundle;

import ale.helper.ALETesterECReportSpec;
import ale.helper.ALETesterECSpec;

import com.ibm.rfid.epcglobal.ale.soap.
ALEServicePortTypeProxy;
import com.ibm.rfid.epcglobal.ale.soap.
Define;
import com.ibm.rfid.epcglobal.ale.soap.
Subscribe;
import com.ibm.rfid.epcglobal.ale.types.
ECReportSpec;
import com.ibm.rfid.epcglobal.ale.types.
ECReportSpecs;
import com.ibm.rfid.epcglobal.ale.types.
ECSpec;

public class SubscribeWithJMS {

        private static String PROPERTIES_
FILE = "ALEAPITestEtape4";

        public static void main(String[] args)
{
                PropertyResource
Bundle resProp = (PropertyResource
Bundle) PropertyResourceBundle.getBundle
(PROPERTIES_FILE);
                String ALE_URL
= resProp.getString("ALE_URL");
                String notificationURI
= resProp.getString("notificationJMSURI");
                String[] readers =
new String[] { resProp.getString("locationId") };
                String[] includePattern
= new
String[] {resProp.getString("includePattern")};
                String duration
= resProp.getString("duration");
                String specName
= resProp.getString("specName");
                String reportName
= resProp.getString("reportName");
                String reportSet
= resProp.getString("reportSet");


                // Get the ALE
engine proxy
                ALEServicePortType
Proxy proxy = new ALEServicePortTypeProxy();
                proxy.setEndpoint
(ALE_URL);

                // Create an ECSpec
for the specified readers
                ECSpec
ecSpec = ALETesterECSpec.
createECSpec(readers, duration);

                // Get an ECReportSpec
ECReportSpec currentReportSpec
=  ALETesterECReportSpec.createECReport
Spec(reportSet, reportName, includePattern);
                ECReport
Specs reportSpecs = new ECReportSpecs();
                reportSpecs.setReportSpec
(new ECReportSpec[] { currentReportSpec });

                // Add
the ECReportSpecs to the ECSpec
                ecSpec.setReportSpecs
(reportSpecs);
                ecSpec.setIncludeSpecIn
Reports(true);

                // Define and
subscribe the spec to the ALE engine
                Define define =
 new Define();
                Subscribe subscribe =
new Subscribe();
                define.setSpec(ecSpec);
                define.setSpecName
(specName);
                subscribe.setSpecName
(specName);
                subscribe.
setNotificationURI(notificationURI);
                try {
                        proxy.
define(define);
                        proxy.
subscribe(subscribe);

                System.out.println
("ECSpec successfully defined
and subscribed");
                } catch (Exception e) {
                System.out.println
("Exception:
" + e.getMessage());
                }
        }
}
```

avec le fichier de propriété ALEAPTestEtape4 :

```
ALE_URL=http://9.212.11.71:
9080/CandidateAleSoapService/
services/ALEServicePort
notificationJMSURI=jms://ibm.rfid.qm/
ale.report.q
locationId=Porte1,Porte2
duration=5000
includePattern=urn:epc:pat:sscc-96.*.*.*
specName=CurrentSetSpec
reportName=PalletTagsReport
reportSet=ADDITIONS
```

Que l'on reçoive les rapports ALE par notification ou lecture dans une queue JMS, les informations fournies suite à la lecture d'étiquettes le seront sous forme de document XML contenant un header et les tags lus. Par exemple, si deux palettes sont chargées, le rapport reçu à l'étape 4 pourrait être :

```
...
<report reportName="PalletTagsReport">
<group>
<groupList>
<member><tag>urn:epc:tag:
sscc-96:21.30.2000</tag></member>
<member><tag>urn:epc:tag:
sscc-96:21.30.2001</tag></member>
</groupList>
</group>
...
```

Pour déployer avec succès une solution RFID, il est donc important de ne pas négliger l'apport d'un middleware. Il apporte :

- des fonctions de filtrage et d'agrégation des lectures RFID effectuées par les lecteurs
- une configuration centralisée de la topologie des lecteurs, contrôleurs de points de lecture permettant de se concentrer sur le développement de la logique métier de la solution cible.

Franck Boudinet
I/T architect,

Didier Decroos
I/T specialist

Joel Viale
I/T specialist du **Centre IBM Européen de solutions** RFID de la Gaude

Le magazine qui parle votre langage

3 Numéros GRATUITS

En ligne
www.programmez.com

- 31%

Un an de veille technologique et d'information permanente, en bénéficiant de

20,45 €
d'ÉCONOMIE
(Prix pour la France Métropolitaine)

Ne manquez plus aucun numéro, abonnez-vous aujourd'hui !

45 € seulement !
Au lieu de 65 €, prix de vente au numéro.
(Prix pour la France Métropolitaine)

OFFRES SPÉCIALES : 39€

• ÉTUDIANTS :
joignez à votre abonnement la photocopie de votre carte d'étudiant (abonnement par voie postale)

• ÉQUIPES :
à partir de 3 abonnements dans la même société : abonnement en ligne obligatoire

OUI, je m'abonne ! ou abonnement en ligne : www.programmez.com

☐ **ABONNEMENT 1 an (11 numéros)** au prix de 45 € TTC. *Tarif France métropolitaine.*
Tarifs hors France métropolitaine : CEE et Suisse : 51,83 € - Algérie, Maroc, Tunisie : 55,95 € - Canada : 64,33 € - Tom : 79,61 € - Dom : 62,84 € - Autres : nous consulter

☐ **OFFRE ÉTUDIANT 1 an (11 numéros)** au prix de 39 € TTC. ***Offre limitée à la France métropolitaine.***
Photocopie de la carte d'étudiant obligatoire

☐ 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 - 75010 Paris.
abonnements.programmez@groupe-gli.com

Programmez! LE MAGAZINE DU DÉVELOPPEMENT

Offre limitée, valable jusqu'au 1er juin 2006

PROG 86

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.

Zend Framework : un outil prometteur

Depuis l'arrivée de PHP 5, une barrière psychologique a été franchie dans les directions informatiques et l'on voit se généraliser l'utilisation de PHP pour réaliser des applications de plus en plus complexes. La grande force de PHP réside dans son écosystème logiciel. Une multitude d'applications, d'extensions, de classes, de bibliothèques et de frameworks existent, si bien que cela peut parfois tourner au désavantage de PHP.

Dans le monde Java, Struts joue ce rôle fédérateur mais dans le monde PHP, il est bien difficile pour une société désireuse de choisir un de ces frameworks comme plate-forme de développement de s'y retrouver. Ce constat à amené la société Zend (dont les deux créateurs font partie des principaux membres du PHP Group) à profiter de sa notoriété pour lancer un framework dont l'objectif est de devenir la référence en la matière. Nous allons, au travers de quelques exemples, vous présenter différents modules du Zend Framework.

Philosophie du Zend Framework

- Simplicité : PHP est facile à utiliser, aussi il n'est pas question de créer un framework qui le rendrait compliqué. Le Zend Framework se veut simple d'accès et d'utilisation ;

- DRY (Don't Repeat Yourself) : ce concept fait partie intégrante du Zend Framework, il s'agit de définir une action dans un unique endroit ;

- Open Source : le code source du Zend Framework est ouvert et téléchargeable à l'adresse suivante : http://framework.zend.com/download ;

- Qualité : les personnes participant à la création du Zend Framework sont des développeurs reconnus du monde PHP;

- Licence libre permissive : la licence du Zend Framework est simple, vous êtes libres de l'utiliser, de le modifier et de l'inclure dans des produits commerciaux.

- Approche MVC : l'utilisation du design pattern « Modèle Vue Contrôleur » devient de plus en plus générique lors de la conception d'applications Web. L'orientation MVC du Zend Framework prend alors tout son sens.

Installation

Installer Zend Framework est très simple :

- téléchargez l'archive (http://framework.zend.com/) décompressez la sur votre serveur Web.

- Modifiez dans votre php.ini la directive de configuration include_path en y ajoutant le chemin qui mène au dossier du Zend Framework.

Remarque : Vous pouvez aussi changer cette directive de configuration à la volée avec la fonction set_include_path().

Astuce sécurité : Comme toute bibliothèque, il est conseillé d'installer les fichiers dans un dossier non accessible par le serveur Web. Cela permet de limiter les soucis éventuels de sécurité.

Dans sa version actuelle pour utiliser les fonctionnalités MVC, le Zend Framework nécessite un serveur web ayant des fonctionnalités de réécriture d'URL. Si vous utilisez un logiciel tel que wampserver (www.wampserver.com) il vous suffit de décommenter la ligne suivante dans le fichier de configuration httpd.conf et de recharger Apache :

LoadModule rewrite_module modules/mod_rewrite.so

Classe Zend

C'est la classe de base du Zend Framework. Elle regroupe un certain nombre de méthodes statiques permettant notamment de charger des classes, des interfaces ou bien des fichiers. Elle implémente également le Design Pattern Registry qui met à la disposition de l'application un espace où ranger toutes sortes de variables auxquelles on peut accéder à tout moment, de n'importe quel point de l'application. Nous en verrons l'utilité lors de l'utilisation du Design Pattern MVC.

Filtrage des données utilisateurs. La classe Zend_Filter contient des méthodes statiques qui permettent de filtrer des données de plusieurs manières différentes :

- filtrage par liste blanche : on teste une variable pour voir si elle est bien au format voulu

- filtrage par liste noire : on retire des données d'une variable, par exemple les balises HTML

- filtrage à l'aveugle : on récupère uniquement certaines données d'une variable, par exemple uniquement les chiffres, ou uniquement les caractères alphabétiques

La classe Zend_InputFilter est une extension de Zend_Filter créée pour travailler plus spécifiquement sur des tableaux et plus particulièrement sur les super globales $_GET et $_POST. Elle fournit des méthodes de filtrage des données provenant des utilisateurs. Toutes les méthodes de cette classe obéissent à une règle simple : si la donnée à filtrer est valide, elle est retournée, sinon FALSE est retourné.

Notez que l'accès direct aux super globales ainsi filtrées n'est pas possible, car Zend_Filter les initialise à NULL une fois le filtrage mis en place. Cela oblige à passer par les méthodes de la classe pour récupérer les données, afin d'éviter les failles de sécurité introduites par inadvertance.

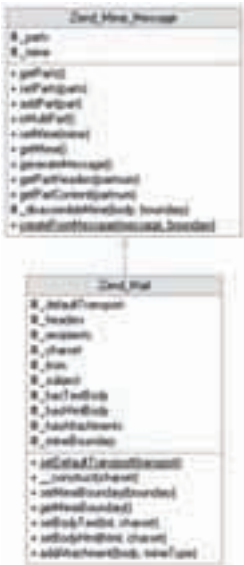
Gestion des logs

Zend_Log est une classe statique générale qui permet de créer et de modifier des fichiers de log. Elle peut être utilisée soit pour un simple log, soit pour plusieurs, et peut éventuellement servir à enregistrer des opérations internes au Zend Framework. Il est possible de paramétrer différents niveaux d'importance de log.

Envoi d'emails

Zend_Mail fournit des fonctionnalités génériques pour envoyer des mails au format texte (ou MIME/HTML via la classe Zend_Mime). Vous pouvez envoyer des mails, soit en utilisant la fonction mail() native de PHP, soit en utilisant une connexion SMTP directe.

Note : pour éviter tout problème de sécurité, Zend_Mail filtre tous les champs d'en-têtes de l'email.
Il est également possible de joindre des fichiers au message.



Mise en place d'une approche MVC

Le Design Pattern MVC acquiert peu à peu ses lettres de noblesse dans le monde des applications PHP dédiées au Système d'Information. Le principe est de diviser une application en trois couches distinctes :

- **La couche modèle** contient des objets chargés de manipuler les données (il s'agit de la logique de votre application),
- **la couche vue** permet l'affichage des écrans (il s'agit de l'interface graphique de votre application),
- **la couche contrôleur** coordonne l'ensemble de l'application et gère les interactions avec l'utilisateur.

Zend Framework permet de construire les applications selon ce Design Pattern. Pour ce faire, nous disposons des classes Zend_Controller et Zend_View. La gestion des modèles est laissée à la guise du développeur. Celui-ci peut, par exemple, utiliser les classes de base de données pour construire les modèles.

Partie contrôleur

La gestion de la couche contrôleur dans le Zend Framework est assez poussée, mais il n'est pas nécessaire de la maîtriser dans son intégralité pour construire une application simple. Avant toute chose, il faut comprendre comment le framework analyse et gère les demandes des utilisateurs. L'URL doit se conformer à une norme afin de pouvoir être analysée et interprétée correctement. Voici comment est construite une URL valide :
http://www.exemple.com/contrôleur/action/cle1/valeur1/

On voit que la partie suivant immédiatement le nom de domaine est le nom du contrôleur, suivie de l'action à effectuer puis des ensembles de clés et de valeurs. Ces ensembles peuvent être multipliés si nécessaire. Le nom du contrôleur et de l'action peut être omis. Dans ce cas, la valeur par défaut « index » sera utilisée.
MVC implique d'avoir une seule porte d'entrée pour l'application afin de répartir les demandes des utilisateurs vers le contrôleur adéquat. A l'heure actuelle, il faut passer par une étape de réécriture d'URL. Cette technique peut effrayer mais dans le cadre du Zend Framework, elle consiste simplement à ajouter un fichier .htaccess (à la racine de l'application si vous utilisez Apache). Ce fichier doit contenir les lignes suivantes :

```
RewriteEngine on
RewriteRule !\.(js|ico|gif|jpg|png|css)$ index.php
```

Remarque : Il faut aussi que la directive de réécriture d'url soit activée dans le httpd.conf
La première ligne active la réécriture d'URL tandis que la seconde indique au serveur de renvoyer toutes les demandes ne concernant pas un fichier avec l'une des extensions listées, vers index.php. Il est également possible de faire la même chose avec d'autres serveurs HTTP qui permettent la réécriture d'URL en adaptant la syntaxe.
Toujours par souci de standardisation, le Zend Framework préconise d'adopter la structure suivante pour une application :

```
/application
    /models
    /views
    /controllers
/document_root
    /images
    /styles
    .htaccess
    index.php
/library
    /Zend
```

Une application nécessite au minimum un contrôleur nommé IndexController. Celui-ci doit se trouver dans le fichier IndexController.php situé dans le répertoire controllers.

Voici comment on déclare un contrôleur minimal :

```
<?php
require_once 'Zend/Controller/Action.php';
class IndexController extends Zend_Controller_Action
{
        public function indexAction()
        {
                echo 'Le contrôleur de base';
        }
        public function noRouteAction()
        {
                $this->_redirect('/');
        }
}
?>
```

La méthode « indexAction » correspond à l'action par défaut « index » qui doit être présente dans tout contrôleur. Passons maintenant à la gestion des vues.

Partie vue

La partie vue de MVC n'est pas liée à un moteur de templates dans le Zend Framework. Partant du principe que PHP est lui-même un moteur de templates, certes évolué, les vues ne sont que des fichiers HTML, avec un peu de PHP pour les boucles, les tests et les affichages. Prenons l'exemple de l'affichage d'une liste de disques. Voici à quoi ressemble le code PHP présent dans le contrôleur :

```
<?php
// Récupération des données depuis un modèle
$disques = new Disques;
// Instanciation et configuration de la vue
Zend::loadClass('Zend_View');
```

```
$vue = new Zend_View;
$vue->setScriptPath('chemin/vers/les/vues);
$vue->disques = $disques->lister();
// Affichage de la vue
echo $vue->render('disques.php');
?>
```

Quant au script de vue, voici à quoi il pourrait ressembler :

```
<?php if ($this->disques): ?>
    <table>
        <tr>
            <th>Artiste</th>
            <th>Titre</th>
        </tr>

        <?php foreach ($this->disques as $disque): ?>
        <tr>
            <td><?php echo $this->escape($disque
['auteur']) ?></td>
            <td><?php echo $this->escape
($disque['titre']) ?></td>
        </tr>
        <?php endforeach; ?>
    </table>
<?php else: ?>
    <p>Il n'y a aucun disque disponible.</p>
<?php endif; ?>
```

Pour éviter d'avoir à instancier et configurer la vue dans chaque contrôleur, il est possible de le faire une fois pour toute dans le fichier index.php, par exemple, et de stocker l'instance dans le registre. Ce dernier est une implémentation du Design Pattern Registry qui permet de stocker puis de récupérer simplement tout objet ou variable que l'on souhaite rendre disponible à l'ensemble de l'application.

```
// Instantiation et configuration dans le fichier index.php
Zend::loadClass('Zend_View');
$vue = new Zend_View;
$vue->setScriptPath('chemin/vers/les/vues');
Zend::register('vue', $vue);
// Utilisation dans un controlleur
$vue = Zend::registry('vue');
$vue->render('script.php');
```

Zend Framework ne fournit pas de moteur de templates. Néanmoins, sa structure ouverte fait qu'il est très simple de substituer son moteur de templates préféré au système de vues fourni. Voici comment utiliser le célèbre Smarty avec l'exemple des disques :

```
<?php
// Récupération des données depuis un modèle
$disques = new Disques;

// Instanciation et configuration de la vue
include 'smarty/Smarty.class.php';
$vue = new Smarty;
$vue->template_dir = 'chemin/vers/les/vues';
$vue->compile_dir = 'chemin/vers/le/cache';
$vue->assign('disques', $disques->lister());
// Affichage de la vue
$vue->display('disques.tpl');
?>

{if $disques}
    <table>
        <tr>
            <th>Artiste</th>
            <th>Titre</th>
        </tr>

        {foreach from=$disques item=disque}
        <tr>
            <td>{$disque.auteur|escape}</td>
            <td>{$disque.titre|escape}</td>
        </tr>
        {/foreach}
    </table>
{else}
    <p>Il n'y a aucun disque disponible.</p>
{/if}
```

Bien entendu, l'astuce consistant à mettre l'instance du moteur de vues dans le registre de l'application fonctionne également avec Smarty.

Accès aux bases de données

Zend Framework fournit une couche d'abstraction de bases de données. A noter qu'il est cependant tout à fait possible d'utiliser une librairie externe comme PEAR DB ou ADOdb. La couche d'abstraction de Zend Framework utilise PDO, l'extension PHP intégrée à PHP 5.1 et disponible via PECL à partir de la version 5.0.3. PDO fournit une API pour accéder à de multiples SGBD comme MySQL, Oracle, MS SQL Server, PostgresSQL ou encore SQLite. Parmi les fonctionnalités offertes par la couche d'abstraction du Zend Framework, on trouve un mécanisme de protection contre les injections SQL, la gestion des transactions ainsi qu'un ensemble de méthodes pour ajouter, modifier ou supprimer simplement des enregistrements.

```
require_once 'Zend/Db.php';
$params = array (
    'host'     => '127.0.0.1',
    'username' => 'root',
    'password' => 'mot_de_passe',
    'dbname'   => 'mon_application'
);
$bdd = Zend_Db::factory('pdoMysql', $params);
// Ajout
$personne = array (
    'nom'    => 'Dupont',
    'prenom' => 'Jean',
    'email'  => 'jean.dupont@exemple.com'
);
```

```
$bdd->insert('personnes', $personne);
$id = $bdd->lastInsertId();
// Modification
$champs = array (
    'email' => 'dupont.jean@exemple.com'
);
$where = $bdd->quoteInto('id = ?', $id);
$enregistrements_affectes = $bdd->update('personnes',
$champs, $where);
// Suppression
$where = $bdd->quoteInto('id = ?', $id);
$enregistrements_affectes = $bdd->delete('personnes', $where);
```

La sélection d'enregistrements se fait à l'aide des méthodes dont le nom commence par fetch. Chacune permet de récupérer le résultat sous une forme directement exploitable en fonction du besoin.

```
// Retourne tous les champs de tous les enregistrements sous forme
de tableau indexé
$resultat = $bdd->fetchAll(
    "SELECT * FROM personnes WHERE prenom = :prenom",
    array('prenom' => 'Jean')
);
// Retourne tous les champs de tous les enregistrements sous forme
de tableau associatif

// dont la clé est le premier champ
$resultat = $bdd->fetchAssoc(
    "SELECT * FROM personnes WHERE prenom = :prenom",
    array('prenom' => 'Jean')
);

// Retourne le premier champ de tous les enregistrements
$resultat = $bdd->fetchCol(
    "SELECT nom FROM personnes WHERE prenom = :prenom",
    array('prenom' => 'Jean')
);

// Retourne le premier champ du premier enregistrement
$resultat = $bdd->fetchOne(
    "SELECT COUNT(*) FROM personnes WHERE prenom
= :prenom",
    array('prenom' => 'Jean')
);
// Retourne un tableau associatif dont la clé est le premier champ
et la valeur le second
$resultat = $bdd->fetchPairs(
    "SELECT nom, prenom FROM personnes WHERE prenom
= :prenom",
    array('prenom' => 'Jean')
);

// Retourne tous les champs du premier enregistrement
$resultat = $bdd->fetchRow(
    "SELECT * FROM personnes WHERE prenom = :prenom",
    array('prenom' => 'Jean')
);
```

Une application est parfois amenée à être utilisée avec plusieurs SGBD. La classe Zend_Db_Select permet de construire des requêtes SQL avec une syntaxe objet. Le code SQL adéquat sera généré en fonction du SGBD.

```
$select = $bdd->select();

$select->from('personnes', '*');
$select->where('prenom = ?', 'Jean');
$select->order('nom');
$select->limit(0, 20);
$personnes = $bdd->fetchAll($select);
```

Conclusion

S'il possède de nombreuses qualités et des fonctionnalités de haut niveau intéressantes (génération de fichiers Pdf, outil de recherche, utilisation de Web Services), le Zend Framework souffre également de quelques défauts, notamment l'absence d'un gestionnaire de formulaires. Il ne faut cependant pas perdre de vue que la version actuellement disponible du Zend Framework n'est qu'une version de développement. La version finale est prévue d'ici fin 2006. Certains aspects sont donc encore un peu bruts, voire absents, mais les choix de conception laissent penser que le produit final sera de bonne qualité.

Soyons francs, Zend Framework n'a rien de révolutionnaire. Mais dans le fond, il n'y a rien d'étonnant à cela. Son but est de fournir un socle robuste pour le développement d'applications modernes, mais sans contraindre les développeurs plus que nécessaire. Il vient donc se placer entre les frameworks qui fournissent tout ou presque, mais manquent de souplesse, et les sites développés totalement à la main. La frontière est fine et souvent, le framework est vu à la fois comme trop contraignant pour les partisans du fait main et comme offrant trop peu de fonctionnalités de haut niveau pour les partisans des frameworks poussés. C'est un positionnement osé, mais probablement nécessaire pour s'adapter à un maximum de situations. La clé du succès sera de réussir à permettre l'intégration aisée des outils PHP existants (moteurs de templates, packages PEAR, libraires PHP, etc.) afin de fédérer le meilleur du monde PHP autour de ce framework.

Jean Marc FONTAINE et Jérôme RENARD
Avec la participation de Cyril PIERRE de GEYER

Tests unitaires avec Nunit

Un développeur ne peut jamais prouver que son programme est sans erreur. Et comme il ne peut tester tout le temps, ni tous les cas possibles, il existe des solutions pour automatiser ces tests. Nunit en est une. Visual Studio 2005 Team System en est une autre...

L'analyse des besoins et les tests représentent plus de 50 % du coût de développement. Si vous voulez diminuer le coût d'un développement tout en augmentant la satisfaction des utilisateurs finaux, vous devez prévoir des tests. Pour réaliser cet objectif, sans pour autant fournir plus d'efforts, vous devez utiliser un framework de tests. Il est généralement admis qu'un jeune diplômé sur trois commence par pratiquer le test. Ajoutons qu'il est aussi de notoriété publique qu'un projet sur deux échoue en raison de mauvais tests (pas de non régression, ou maintenance difficile, ou trop grand nombre de bogues). Maintenant, soyons sincères : le nombre de programmeurs qui testent en utilisant un framework comme Nunit sont encore trop peu nombreux. Mais dans une certaine mesure, cela va certainement changer sous .Net, avec l'arrivée de Visual Studio 2005 Team System qui incorpore son propre framework de tests unitaires.

Qu'est ce qu'un test unitaire ?

Le test unitaire est aussi appelé test de composant. Il a pour but de contrôler la fiabilité des « unités logicielles » développées. Il s'attache à vérifier le respect des spécifications fonctionnelles en détectant les erreurs de logique. De manière pratique pour chaque unité, le programmeur doit effectuer un test en se servant des données en entrées pour observer les sorties et son fonctionnement. Par unité, dans le contexte de la programmation objet, il faut comprendre une classe.

Si le test unitaire échoue, il y a lieu de modifier le code pour qu'il réponde à la spécification. Si le test réussit, il devra être rejoué après chaque ajout/modification du code pour vérifier que cet ajout n'a pas entraîné l'apparition de nouvelles erreurs (c'est ce que l'on appelle les tests de non régression). Classiquement, le code est rédigé puis les tests sont écrits, mais le développeur peut aussi commencer par écrire les tests et ensuite construire son code, ou bien utiliser une technique intermédiaire : au fur et à mesure de l'avancement du projet, les tests sont rédigés. Les tests unitaires sont fortement influencés par le courant « Xtreme programming » qui met en avant le « Parallel Peer Programming ». En résumé, cette technique implique une session de travail en parallèle de deux programmeurs : le premier écrit les tests unitaires, et l'autre écrit le code pour passer les tests.

Le Parallel Peer Programming de l'Xtreme programming

- Les tests unitaires sont rédigés en parallèle avec le programme ;
- Les tests unitaires décrivent le comportement de l'application ;
- Si le test unitaire a été écrit avant son code, le test échoue ;
- Le développeur de l'application programme pour faire en sorte que son logiciel passe tous les tests automatiques. Il n'a pas besoin d'en « faire plus » : si le test passe, plus besoin de coder.

1 Installation de Nunit

Nunit est un framework de tests unitaires Open Source. Il tourne sous Windows et sous Linux avec Mono. Sous Linux, nous allons commencer par piocher la dernière version de Mono (en l'occurrence, pour nous, la version 1.1.13). Pour y parvenir nous avons utilisé Red Carpet de Novell (rpm -Uvh rcd-2.4.9-2.i586.rpm ; rpm -Uvh rug-2.4.9-2.i586.rpm ; /etc/init.d/rcd start).

```
#/etc/init.d/rcd status
Checking for Red Carpet Daemon:                    running
```

Enfin, nous devons installer au minimum mono-basic et mono-nunit (rug install mono-basic ; rug install mono-nunit). Ce qui donne :

```
# rug search mono
```

S	Channel	Name	Version
	Mono 1.1.x	mod_mono	1.1.13-0.suse100.novell
i	Mono 1.1.x	mono-basic	1.1.13.2-0.novell
	Mono 1.1.x	mono-complete	1.1.13.2-0.novell
i	Mono 1.1.x	mono-core	1.1.13.2-0.novell
i	Mono 1.1.x	mono-data	1.1.13.2-0.novell
	Mono 1.1.x	mono-data-firebird	1.1.13.2-0.novell
	Mono 1.1.x	mono-data-oracle	1.1.13.2-0.novell
	Mono 1.1.x	mono-data-postgresq	1.1.13.2-0.novell
	Mono 1.1.x	mono-data-sqlite	1.1.13.2-0.novell
	Mono 1.1.x	mono-data-sybase	1.1.13.2-0.novell
	Tools for mono	mono-debugger	0.11-0.novell
	Mono 1.1.x	mono-devel	1.1.13.2-0.novell
	Mono 1.1.x	mono-extras	1.1.13.2-0.novell
	Mono 1.1.x	mono-jscript	1.1.13.2-0.novell
	Mono 1.1.x	mono-locale-extras	1.1.13.2-0.novell
i	Mono 1.1.x	mono-nunit	1.1.13.2-0.novell
	Tools for mono	mono-tools	1.1.11-0.novell
i	Mono 1.1.x	mono-web	1.1.13.2-0.novell
i	Mono 1.1.x	mono-winforms	1.1.13.2-0.novell
	Tools for mono	monodevelop	0.9-2.novell
	Tools for mono	monodoc-core	1.1.13-0.novell

Sous Windows, NUnit 2.2.7 est téléchargeable en version .Net 2.0 ou 1.1 (Nunit-2.2.7-net-1.1.msi ou Nunit-2.2.7-net-2.0.msi).

2 Mise au point du logiciel sans les tests

Notre intention est de demander une adresse IP à l'utilisateur pour, par exemple, ensuite établir une liaison sécurisée ou réaliser un simple ping. En C# nous pourrions commencer par écrire :

```
demandeip.cs
using System;
public class ProgCS
{
  static void Main()
 {
   string strAdresse = "";
   Console.WriteLine("Encodez une adresse IP valide : ");
   strAdresse = Console.ReadLine();
   Console.WriteLine("Traitement sur : " + strAdresse);
  }
}
```

```
#mcs demandeip.cs
#mono demandeip.exe
```

Seulement, rappelons qu'il s'agit de tester des classes. Le code source tel qu'il a été écrit ne convient donc pas. Nous allons devoir rendre accessible la donnée strAdresse via une classe AdresseIp en utilisant les accesseurs get et set. Ceux-ci contiendront des instructions associées à l'obtention (lecture get), ou à la définition de l'adresse IP (écriture set).

```
adresseip_sanstests.cs
using System;

public class AdresseIp
{
    private string Adresse;
    public AdresseIp(string Adresse)
    {
        this.Adresse = Adresse;
    }
    public string Adr
    {
      get{return Adresse;}
      set{this.Adresse = value;}
    }
}

public class ProgCS
{
  static void Main()
  {
    string strAdresse = "";
    AdresseIp adr = new AdresseIp(strAdresse);
    Console.WriteLine("Encodez une adresse IP valide : ");
    strAdresse = Console.ReadLine();
    adr.Adr = strAdresse;
    Console.WriteLine("Traitement sur : " + adr.Adr);
  }
}
```

Compilons :

```
#mcs adresseip_sanstests.cs
```

Puis exécutons :

```
#mono adresseip_sanstests.exe
Encodez une adresse IP valide :
1920.168
Traitement sur : 1920.168
```

Le code de base ne contient aucun test. Le logiciel accepte, par exemple, que l'adresse encodée soit 1920.168, ce qui ne veut rien dire.

3 Mise au point des tests

Attention, et c'est là un point important, nous n'allons pas réfléchir au «comment», mais établir un certain nombre de règles que devra respecter une adresse IP. Par exemple, nous acceptons que 192.168.0.10 (test 1) soit une adresse ip valide, tandis que nous rejetons 192.168.10 (test2). Une adresse ip vide (test3) ou une adresse ip 0.0.0.0 n'est pas non plus acceptable (test 4). Concrètement, nous pouvons établir un petit tableau présentant les adresses ip testées en entrée et le résultat attendu en sortie :

Entrée	Test	résultat
192.168.0.10	=>	vrai
192.168.0	=>	faux
ND	=>	faux
0.0.0.0	=>	faux

On peut maintenant considérer que vrai « n'est pas faux », et que «faux» sera un message d'erreur (« ERREUR »), produit par la routine de validation qu'il faudra programmer. A cet effet, nous utilisons l'espace de nom Nunit.Framework. La classe de test est marquée de l'attribut [TestFixture] qui indique au framework NUnit sa nature de classe de test. Nous avons séparé les tests de chaque fonctionnalité dans des méthodes séparées (test IP1 à test IP4), mais ce n'est pas obligatoire, quoique vivement conseillé pour obtenir une meilleure lisibilité. Chaque méthode devant être testée sera marquée de l'attribut [Test] (sinon il n'y a pas d'exécution). Ces tests sont appelés « cas d'essais unitaires ».

```
testip.cs
using System;
using NUnit.Framework;
using System.Text.RegularExpressions;

[TestFixture]
public class testip
{
   [Test]
    public void TestIP1()
    {
        string testadresse = "192.168.0.10";
        AdresseIp adr = new AdresseIp(testadresse);
        Assert.IsFalse(adr.Adr=="ERREUR");
    }
    [Test]
    public void TestIP2()
    {
        string testadresse = "192.168.10";
        AdresseIp adr = new AdresseIp(testadresse);
        Assert.AreEqual("ERREUR",adr.Adr);
```

```
    }
    [Test]
    public void TestIP3()
    {
        string testadresse = "";
        AdresseIp adr = new AdresseIp(testadresse);
        Assert.AreEqual("ERREUR",adr.Adr);
    }
    [Test]
    public void TestIP4()
    {
        string testadresse = "0.0.0.0";
        AdresseIp adr = new AdresseIp(testadresse);
        Assert.AreEqual("ERREUR",adr.Adr);
    }
}
```

Pour compiler les tests avec la classe à tester nous devons référencer « nunit.framework.dll » :

#mcs adresseip_sanstests.cs testip.cs -r:nunit.framework.dll

Lançons les tests unitaires sans avoir programmé la validation de l'adresse ip :

#mono /usr/lib/mono/1.0/nunit-console.exe adresseip_sanstests.exe

```
NUnit version 2.2.0
Copyright (C) 2002-2003 James W. Newkirk, Michael C. Two, Alexei A.
Vorontsov, Charlie Poole.
Copyright (C) 2000-2003 Philip Craig.
All Rights Reserved.

OS Version: Unix 2.6.13.15    Mono Version: 1.1.4322.2032

..F.F.F
Tests run: 4, Failures: 3, Not run: 0, Time: 0.304787 seconds
Tests run: 4, Failures: 3, Not run: 0, Time: 0.304787 seconds

Failures:
1) testip.TestIP2 :
        String lengths differ.  Expected length=6, but was length=10.
        Strings differ at index 0.

        expected:<"ERREUR">
         but was:<"192.168.10">
        ____^

in <0x0003d> testip:TestIP2 ()
in <0x00000> <unknown method>
in (wrapper managed-to-native)
 System.Reflection.MonoMethod:InternalInvoke (object,object[])
in <0x0008d> System.Reflection.MonoMethod:Invoke (System.Object
obj, BindingFlags invokeAttr, System.Reflection.Binder binder,
 System.Object[] parameters, System.Globalization.CultureInfo culture)
...
```

Trois tests sur quatre échouent. Par exemple, au deuxième test, le framework attend « ERREUR » mais le résultat produit en sortie est 192.168.10.

4 Correction du code afin de passer les tests unitaires

Nous allons employer le moteur d'expressions régulières de .Net pour valider l'adresse IP (instance de la classe Regex de l'espace de nom « System.Text.RegularExpressions »). Pour valider une adresse IP, nous créerons une nouvelle expression régulière « maregexp » précédée de « @ » pour éviter la production par le compilateur de l'erreur « séquence d'échappement non reconnue ». La méthode resultat.Success validera ou non l'expression. Enfin, nous avons ajouté un test invalidant l'adresse ip spéciale « 0.0.0.0 » qui est, par contre, acceptée par l'expression régulière.

Voici ce que donne notre test de validation :

```
adresseip.cs
using System;
using System.Text.RegularExpressions;

public class AdresseIp
{
    private string Adresse;

    public AdresseIp(string Adresse)
    {
        this.Adresse = Adresse;
    }

    public string Adr
    {
    set{
        this.Adresse = value;
    }
    get{
        Regex maregexp = new Regex(@"^(([01]?\d\d?|2[0-4]\d|25
[0-5])\.([01]?\d\d?|2[0-4]\d|25[0-5])\.([01]?\d\d?|2[0-4]\d|25
[0-5])\.([01]?\d\d?|2[0-4]\d|25[0-5]))$");
        Match resultat = maregexp.Match(Adresse);
        if (resultat.Success) {
            if (Adresse == "0.0.0.0") {
                return "ERREUR";
            }
            else
                return Adresse;
        }
        else
            return "ERREUR";
        }
    }
}

public class ProgCS
{
  static void Main()
  {
    string strAdresse = "";
    AdresseIp adr = new AdresseIp(strAdresse);
```

```
    while (adr.Adr == "ERREUR") {
        Console.WriteLine("Encodez une adresse IP valide : ");
        strAdresse = Console.ReadLine();
        adr.Adr = strAdresse;

        if (adr.Adr != "ERREUR")
            Console.WriteLine("Traitement sur : " + adr.Adr);
        else
            Console.WriteLine("Merci de bien vouloir encoder une adresse
 IP valide !");
    }
  }
}
```

Compilons cette correction de code :

mcs adresseip.cs testip.cs -r:nunit.framework.dll

Et testons ce nouvel assemblage dynamiquement :

mono /usr/lib/mono/1.0/nunit-console.exe adresseip.exe

```
NUnit version 2.2.0
Copyright (C) 2002-2003 James W. Newkirk, Michael C. Two, Alexei A.
Vorontsov, Charlie Poole.
Copyright (C) 2000-2003 Philip Craig.
All Rights Reserved.
OS Version: Unix 2.6.13.15    Mono Version: 1.1.4322.2032
....
Tests run: 4, Failures: 0, Not run: 0, Time: 0.466157 seconds
```

Le résultat « passe », ce qui veut dire que tant que de nouvelles spécifications ne sont pas ajoutées (un test 5 par exemple qui interdirait l'adresse 255.255.255.255), le code est correct et restera inchangé. Par contre, vous devrez relancer régulièrement les tests unitaires pour vérifier la non régression (test de non régression) au fur et à mesure de l'état d'avancement global de l'application (test d'intégration). En clair, vous devez vérifier à chaque modification d'une partie du code que « ce qui marchait, marche encore ». Et même après développement, en phase de maintenance, les tests resteront primordiaux : par exemple, en cas de refactoring, ou d'ajout ou de suppression de fonctionnalités, ou après la correction d'un bogue.

5 Nunit et Visual Studio 2005 Team System

Pour commencer, il existe un Add-in pour Visual Studio permettant d'utiliser le framework NUnit à partir de l'environnement de développement de VS (TestDriven.NET-2.0.1438d). Cependant, il faut savoir que Jim Newkirk, un des concepteurs de NUnit, a développé un convertisseur de cas d'essais unitaires NUnit en cas d'essais unitaires, compatible avec Visual Studio 2005 Team System (NUnit Converter V1.0 RC1).

Si vous avez Visual Studio 2005 Team System (VSTS) vous pourrez convertir vos projets Nunit et abandonnerez probablement Nunit. C'est principalement une question d'intégration. Pas besoin d'add-in spécialisé. Avec VSTS vous gagnez un outil de génération automatique des tests unitaires. Ce qui est génial, c'est que d'un seul clic de souris vous pouvez sélectionnez les méthodes à tester pour en générer automatiquement le code d'appel des méthodes. Restera pour le développeur à spécifier les valeurs en entrées et celles attendues en sorties (en indiquant avec une

Outil de couverture de code intégré dans Visual Studio 2005 Team System

syntaxe d'assertions si le test est réussi). Mieux, vous pourrez importer un jeu de données externe (par exemple sous la forme d'un fichier Excel). En outre, VSTS incorpore un outil de couverture de code.

En clair, le développeur peut obtenir des résultats statistiques permettant de constater quelles sont les proportions du code qui ont été testées. Le programmeur peut ainsi visualiser le code qui a été testé et celui qui ne l'a pas été (directement dans l'éditeur de code, par affichage d'une couleur différente pour les lignes testées et non testées !).

Mise en évidence en vert du code testé ou non testé en (rouge).

Liste des liens intéressants :

Nunit : http://www.nunit.org

rpm rcd pour SuSE 10 :

http://primates.ximian.com/~rml/sanitize/rcd-2.4.9-2.i586.rpm

rpm rug pour SuSE 10 :

http://primates.ximian.com/~rml/sanitize/rug-2.4.9-2.i586.rpm

NUnit Converter V1.0 RC1 :

http://www.gotdotnet.com/workspaces/workspace.aspx?id=91936c5e-461f-4027-bdba-8a46f52fefdb

Addin Nunit pour VS : http://www.testdriven.net

■ **Xavier Leclercq**
Xavier.Leclercq@programmez.com

Sécuriser un service Web avec les WSE 3.0

Microsoft propose nativement dans .Net des classes permettant de développer des Web Services. Cependant, si vous souhaitez construire des WebServices sécurisés respectant les standards WS-, il vous faudra utiliser les Web Services Enhancements. Ils permettent aux développeurs de construire des applications basées sur les dernières spécifications publiées par Microsoft et différents acteurs de l'industrie telles que WS-Security 1.0 et 1.1, WS-Trust, WS-SecureConversation, WS-Addressing et Message Transmission Optimization Mechanism (MTOM).*

Microsoft a accentué ses efforts pour simplifier le développement de services Web sécurisés et leur déploiement. Cette version est une version majeure avant l'arrivée du nouveau modèle de programmation Windows Communication Foundation (WCF). Issues de spécifications de sécurité communes, elles devraient être totalement interopérables. Nous abordons le fonctionnement des WSE 3.0 et plus particulièrement les fichiers de configuration « Policy » ; nous présenterons les nouveautés apportées par ces nouvelles extensions. Enfin, nous terminerons par un exemple de sécurisation d'un service Web via WS-Security et un exemple d'échange de fichiers volumineux avec MTOM.

Le fonctionnement des WSE 3.0

Pour sécuriser un WebServices, WSE 3.0 utilise les mécanismes définis par les spécifications WS-Security et modifie les messages SOAP en y intégrant des informations liées à la sécurité. Une application cliente qui émet un message SOAP peut le sécuriser :

- en ajoutant des informations permettant au destinataire de l'authentifier (nom d'utilisateur/mot de passe, certificat X509),
- en signant le message permettant au destinataire de vérifier qu'il n'a pas été modifié (certificat X509, Ticket Kerberos),
- en cryptant le message afin que seules les personnes autorisées puissent le lire.

WSE 3.0 simplifie la tâche des développeurs, leur permettant de décrire dans un fichier de configuration XML les règles de sécurité à appliquer pour un service Web. Ce fichier, appelons-le « Policy », va contenir un ensemble de stratégies de sécurité; chaque stratégie est identifiée par un attribut « name » qui l'identifie de manière unique. Nous verrons par la suite que nous utiliserons cet attribut dans notre code pour appliquer une stratégie de sécurité à une méthode de notre service Web.

```
<policies>
  <extensions>
    <extension />
  </extensions>
  <policy name="Mastrategie" />
          < UserNameForCertificateSecurity >
                ...
          </ UserNameForCertificateSecurity >
</policies>
```

Regardons à présent la structure du fichier « Policy ». Nous constatons que la stratégie de sécurité contient en exemple une entrée « UserNameForCertificateSecurity ». Cette entrée, dite assertion de sécurité, est utilisée pour décrire les règles de sécurité à appliquer à un message SOAP durant son transfert entre une application cliente et un service Web. WSE 3.0 autorise la création de ses propres assertions de sécurité et en propose six, déjà implémentées sur la base des meilleures pratiques de l'industrie. Chacune répond à sa manière aux principes de sécurité qui sont l'authentification, l'intégrité des messages et la confidentialité.

Les assertions de sécurité prédéfinies sont :

- AnonymousForCertificateSecurity : Le client n'est pas authentifié et le serveur est authentifié avec son certificat X509. C'est le cas d'une transaction d'achat en ligne.
- KerberosSecurity : L'authentification se fait via des tickets Kerberos en environnement Windows.
- MutualCertificate10 : Le client et le serveur s'authentifient en utilisant des certificats X509. Cette assertion repose sur les spécifications WS-Security 1.0.
- MutualCertificate11 : Cette assertion est identique à la précédente, mais repose sur les spécifications WS-Security 1.1.
- UserNameOverTransportSecurity : L'authentification des clients se fait à partir d'un nom d'utilisateur et d'un mot de passe. Le transport est sécurisé via SSL.
- UserNameForCertificateSecurity : L'authentification des clients se fait à partir d'un nom d'utilisateur et d'un mot de passe. Le serveur s'authentifie en utilisant un certificat X509.

Les nouveautés Web Services Enhancements 3.0

Voici quelques une des nouveautés majeures de WSE 3.0 :

Un environnement de développement simplifié

Plusieurs scénarios de sécurisation de WebServices sont proposés

Le modèle objet ainsi que le format du fichier de configuration « Policy » ont été simplifiés par rapport à l'implémentation fournie avec les WSE 2.0.

Ce qui est réalisable via des assistants de manière déclarative est désormais possible par code. Vous pouvez créer des assertions de sécurité en les codant et en utilisant les mêmes paramètres que ceux fournis dans un fichier de configuration « Policy ».

Une intégration à Visual Studio 2005 permettant, via des assistants, de générer automatiquement les fichiers de configuration « Policy ».

• **L'envoi de données binaires volumineuses est facilité**
Les spécifications W3C SOAP MTOM (Message Transmission Optimization Mechanism) sont désormais supportées, permettant le transfert de données binaires volumineuses de façon sécurisée.
• **Le support des derniers standards WS-***
Support des dernières spécifications WS-Security 1.0 et 1.1,WS-Trust, WS-SecureConversation, WS-Adressing
• **L'hébergement de WebServices en dehors d'IIS**
- Il est possible d'héberger un service Web dans une application console ou un service Windows et de l'invoquer via TCP/IP.
• **Le support des plates-formes Windows 64 Bits**
• **L'inter-opérabilité avec les WCF (Windows Communication Foundation)**
Un client WSE3.0 pourra communiquer avec un service WCF et inversement. Un guide de migration de WSE3.0 vers les WCF sera proposé par Microsoft lorsque les WCF seront finalisés.

Installation des WSE 3.0

Le package d'installation des WSE 3.0 (Microsoft WSE 3.0.msi) est téléchargeable depuis le download center de Microsoft. Le menu d'installation propose plusieurs types de mises à jour en fonction des profils utilisateurs. Les développeurs choisiront l'option « Visual Studio Developer » qui installe le runtime, les exemples, la documentation, les outils, et intègre WSE 3.0 dans l'environnement de développement, via des boîtes de paramétrage.

Plusieurs outils sont proposés avec les WSE 3.0 :
- WseWsdl3.exe : Il permet de générer un proxy client à partir d'un fichier WSDL (Web Services Description Language)
- Configuration Tool : Il permet d'activer et de paramétrer les fonctionnalités WSE à partir de l'environnement de développement Visual Studio .Net
- Certificate Tool : Il permet de visualiser les certificats présents sur un ordinateur.

Comment sécuriser un service Web ?

Regardons à présent un exemple qui décrit comment sécuriser un service Web en utilisant une assertion de sécurité de type « UserNameForCertificateSecurity ». Nous allons développer un WebService « WsGestion » qui propose une méthode « SoldeCompte », permettant ainsi de récupérer le solde du compte d'un utilisateur. L'application cliente va invoquer cette méthode et transmettre au service web les informations (nom utilisateur, mot de passe) qui lui serviront pour authentifier l'utilisateur. Si celui-ci fait partie du groupe « Administrateur », alors il aura accès au solde de son compte.

La sécurisation du WebService :

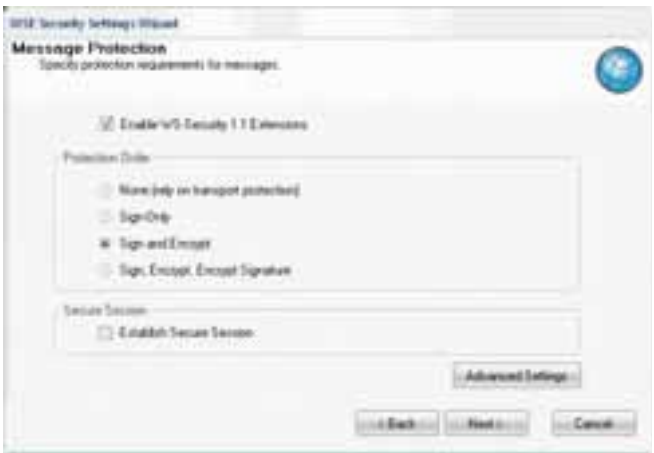
```
using Microsoft.Web.Services3;
using Microsoft.Web.Services3.Security;
using Microsoft.Web.Services3.Security.Tokens;

...
    [WebMethod]
    public string SoldeCompte() {
        try
        {
            // Vérification des droits de l'utilisateur
            VerifierPermission();

            // Accès à la base de données et récupération
du solde de l'utilisateur
            // ...

            return "Le solde du compte est : 2500 ?";
        }
        catch (UnauthorizedAccessException ex)
        {
            throw ex;
        }
    }

    private void VerifierPermission()
    {
        SecurityToken tk = RequestSoapContext.Current.IdentityToken;

        // Vérification que l'utilisateur est bien administrateur et
        // peut accéder au service WsGestion
        bool bRole = tk.Principal.IsInRole(@"Administrateurs");

        if (!bRole)
        {
            throw new UnauthorizedAccessException
("Accès non autorisé au service WsGestion");
        }
    }
}
```

La méthode « SoldeCompte » invoque la méthode « VerifierPermission » afin de vérifier si l'utilisateur est autorisé ou non à récupérer le solde de son compte. La méthode « VerifierPermission » récupère le jeton de sécurité et vérifie si l'utilisateur fait partie du groupe des administrateurs de la machine. Il nous reste à sécuriser ce web service en nous servant de l'assistant WSE 3.0 pour générer automatiquement un fichier de sécurité « Policy ». Pour ce faire, nous sélectionnons l'entrée WSE Setting 3.0 accessible dans les propriétés de notre projet sous Visual Studio 2005.

Puis nous suivons les étapes suivantes :
Etape 1 : Activation des WSE 3.0
Nous activons dans l'onglet « Général » les deux cases « Enable this project for Web Service Enhancements » et « Enable Microsoft Web Service Enhancements Soap Protocol Factory ». La première option active les WSE 3.0 pour le projet, la seconde est nécessaire pour les applications Web.

Etape 2 : Création d'une nouvelle stratégie de sécurité
Nous définissons une nouvelle stratégie de sécurité dans l'onglet « Policy » que nous appelons « PoliceServeur ». Nous sélectionnons la méthode d'authentification « UserName » et indiquons que nous souhaitons sécuriser le service Web.

Nous appliquons la sécurité en demandant la signature et le cryptage de notre message.



Puis nous sélectionnons le certificat X509 qui sera utilisé pour signer le message. Enfin, nous vérifions dans le résumé les options que nous avons sélectionnées. En cliquant sur le bouton « Finish », l'assistant WSE 3.0 va automatiquement générer un fichier de sécurité « Policy ». Celui-ci contient une stratégie de sécurité nommée « PoliceServeur » qui déclare une assertion prédéfinie de type « UserNameForCertificateSecurity ». Notons que cette assertion correspond bien à la méthode d'authentification que nous avions choisie « UserName », le nom est simplement différent.

```
<policies xmlns="http://schemas.microsoft.com/wse/2005/06/
policy">
  <extensions>
…
  </extensions>
  <policy name="PoliceServeur">
    <usernameForCertificateSecurity
establishSecurityContext="false" renewExpiredSecurityContext=
"true" requireSignatureConfirmation=
"false" messageProtectionOrder=
"SignBeforeEncrypt" requireDerivedKeys="true" ttlInSeconds="300">
      <serviceToken>
        <x509 storeLocation="LocalMachine"
storeName="My" findValue="CN=WSE2
QuickStartServer" findType="FindBySubjectDistinguishedName" />
      </serviceToken>
      <protection>
…
      </protection>
    </usernameForCertificateSecurity>
    <requireActionHeader />
  </policy>
</policies>
```

Etape 3 : Configuration du web service pour utiliser la stratégie de sécurité « PoliceServeur ». Pour appliquer la stratégie de sécurité que nous venons de créer à notre web service, il suffit d'ajouter un attribut [Policy] en début de classe avec comme paramètre le nom de la stratégie de sécurité « PoliceServeur ».

```
using Microsoft.Web.Services3;
using Microsoft.Web.Services3.Security;
using Microsoft.Web.Services3.Security.Tokens;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
[Policy("PoliceServeur")]
public class Service : System.Web.Services.WebService
…
```

La sécurisation de l'application cliente :

Pour pouvoir invoquer le web service WsGestion, l'application cliente doit également générer un fichier de sécurité « Policy », comme nous venons de le faire pour la partie serveur. Il faut simplement prendre soin de préciser qu'il s'agit de générer le fichier destiné à l'application cliente. Cette application instancie le proxy permettant l'accès au service web, fournit au serveur un jeton de sécurité définissant le nom et le mot de passe de l'utilisateur et applique la stratégie de sécurité « PoliceClient » en faisant appel à la méthode du framework « SetPolicy ». Enfin, elle invoque la méthode « SoldeCompte ».

```
using Microsoft.Web.Services3.Security.Tokens;
…

wsgestion.ServiceWse  serviceproxy = new wsgestion.ServiceWse();

// Création d'un jeton de sécurité
UsernameToken tk = new UsernameToken("PEYRUSSE","password",
PasswordOption.
SendPlainText);
serviceproxy.SetClientCredential(tk);

serviceproxy.SetPolicy("PoliceClient");
MessageBox.Show(serviceproxy.SoldeCompte());
```

Il est possible d'activer une trace depuis l'onglet Diagnostics et ainsi de capturer, dans des fichiers, les messages SOAP envoyés et reçus par WSE. L'analyse de ces fichiers permet de vérifier que les messages sont bien cryptés et signés automatiquement durant le transfert.

```
…
 <xenc:EncryptedKey Id="SecurityToken-
86a3bdb9-af9a-4699-a11c-fd99d1377b49"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
```

```
            <xenc:
EncryptionMethod Algorithm="http://www.w3.org/2001/04/
xmlenc#rsa-oaep-mgf1p">
            <ds:DigestMethod xmlns:ds="http://www.w3.org/2000/09/
xmldsig#"
 Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
            </xenc:EncryptionMethod>
            <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
              <wsse:SecurityTokenReference>
                <wsse:KeyIdentifier ValueType=
"http://docs.oasis-open.org/wss/oasis-wss-soap-message-
security-1.1#ThumbprintSHA1" EncodingType=
"http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-soap-message-
security-1.0#Base64Binary">XgTT5neAD510ClOx/jg1E9PLybl=
</wsse:KeyIdentifier>
              </wsse:SecurityTokenReference>
            </KeyInfo>
            <xenc:CipherData>
              <xenc:CipherValue>k60MOBD1TBtWy25yaskvef4QofxX/
tHdw0MkFDGyMxDJPLMlXUQpfQk09RHKgmawbA91Oo25qi/
jBjEhDTH3042VOsTdc2PGVNJ/
WnRlWfN678i7KOxLwSxHYHrYxhm2wiuWnueED97
jnY1brjipHObt1YpWWhJC6hEkeO0ZhHU=</
xenc:CipherValue>
            </xenc:CipherData>
          </xenc:EncryptedKey>
…
```

Comment échanger des données binaires avec MTOM

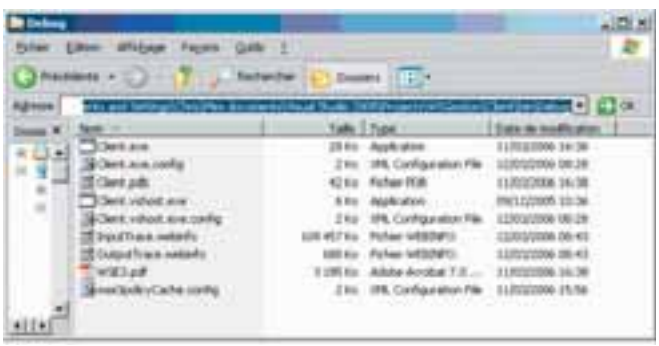
Pour réaliser un transfert de données avec les WSE 2.0, il fallait utiliser un objet de type DimeAttachment pour lire un fichier et le retourner vers une application cliente. Avec les WSE 3.0, ce code doit être migré afin d'utiliser MTOM. Nous allons compléter notre service WsGestion en lui ajoutant une méthode « RetournerFichier ». Cette méthode va à partir du nom d'un fichier passé en paramètre, le lire dans un tableau d'octets (byte[]) et le retourner à l'application cliente.

```
  [WebMethod]
   public Reponse RetournerFichier(string sNomFichier)
   {
      Reponse rep = new Reponse();
      rep.sNomFichier  = sNomFichier;
      String sCheminFichier = AppDomain.CurrentDomain.
BaseDirectory + @"App_Data\" + sNomFichier;
      rep.bContenu = File.ReadAllBytes(sCheminFichier);
      return rep;
   }

[XmlRoot("Reponse", Namespace = "http://tempuri.org/")]
public class Reponse
{
   [XmlElement("NomFichier", IsNullable = false)]
   public string sNomFichier;
   [XmlElement("Contenu", IsNullable = false)]
   public byte[] bContenu;
}
```

L'application cliente ne fait qu'invoquer la méthode « RetournerFichier » du service WsGestion et lui demande le transfert du fichier appelé « WSE3.pdf ». Une fois réceptionnée, elle affiche son nom ainsi que sa taille.

```
using System.IO;
…

String sNomFichier = "WSE3.pdf";

wsgestion.ServiceWse serviceproxy = new wsgestion.ServiceWse();

// Création d'un jeton de sécurité
UsernameToken tk = new UsernameToken
("PEYRUSSE", "password",PasswordOption.SendPlainText);
serviceproxy.SetClientCredential(tk);
serviceproxy.SetPolicy("PoliceClient");

wsgestion.Reponse rep = serviceproxy.RetournerFichier(sNomFichier);

System.Diagnostics.Debug.Write("Nom du fichier : "
+ rep.NomFichier + "\n");
System.Diagnostics.Debug.Write("Nombre d'octets reçus:
" + rep.Contenu.Length);
File.WriteAllBytes(rep.NomFichier, rep.Contenu);
```

Regardons à présent le résultat de l'exécution.



L'application cliente a bien réceptionné le fichier de plus de 3 Mo copié dans le répertoire courant. Elle affiche dans la fenêtre debug le nom et la taille du fichier.

```
Nom du fichier : WSE3.pdf
Nombre d'octets reçus: 3260527
```

Conclusion

Nous avons montré quelques unes des nouveautés des WSE 3.0. Cette nouvelle version a été simplifiée et offre de nouvelles fonctionnalités qui facilitent la mise en œuvre de la sécurité pour les développeurs et les administrateurs. Outre le respect des derniers standards WS-*, l'adopter est un investissement pour l'avenir et vous permettra d'être mieux préparés lorsque les WCF arriveront ; les deux technologies étant compatibles.

■ Christian PEYRUSSE

Le singleton, un objet unique en son genre

Un singleton est une classe qui ne peut être instanciée qu'une seule fois. Une idée simple, mais pas nécessairement simple à appliquer. Discutons en ensemble autour d'une tasse de Java.

On déclare fièrement que Java est un langage plus simple et plus sûr que C++. C'est effectivement le cas, mais ce serait un excès d'optimisme de penser que Java est toujours facile. Comme les autres, ce langage est une boîte à surprises. Ainsi nous allons voir aujourd'hui qu'écrire un singleton en Java est un véritable exercice. Mais avant tout, qu'est-ce donc qu'un singleton et à quoi cela sert-il ? Un singleton est un objet instancié une fois et une seule. Un tel objet a son utilité dans nombre de cas. Par exemple, un spooler d'impression, un gestionnaire de fenêtre, un équilibreur de charges, un pilote. Ou encore si on veut éviter des créations d'objet à répétition, coûteuses dans un environnement embarqué. Les cas sont extrêmement nombreux. Voici deux exemples de singletons que vous avez probablement rencontrés dans votre JVM : java.awt.Toolkit et java.lang.Runtime. L'expérience des programmeurs aidant, le singleton est devenu un modèle de conception, un Design Pattern comme on dit en programmation objet. Tout Design Pattern qu'il est, le singleton fait encore beaucoup discuter en programmation Java. On voit souvent dans les livres ou sur Internet des recommandations pas toujours raisonnables. Essayons-nous à l'exercice du singleton Java.

1 Le tout static

L'instanciation unique du singleton est un problème ? Qu'à cela ne tienne, il n'y a qu'à écrire une classe dans laquelle tout est statique. Plus besoin d'instancier du tout et tout est réglé. Telle est l'opinion des partisans du singleton "tout static". Effectivement, dans certaines situations, la démarche peut être adaptée. Toutefois, nous pensons qu'il s'agit d'un style qui rappelle plus la programmation spaghetti d'un autre âge que la programmation objet. Et c'est là qu'est le problème. Avec le "tout static," le polymorphisme est cassé et notre singleton ne pourra pas être une classe dérivée. Même remarque en ce qui concerne les interfaces : impossible d'en implémenter une en "tout static". Enfin, si le code doit être repensé et le singleton abandonné, remanier une classe "tout static" demandera sans doute plus de travail. Nous réservons donc cette solution à des cas très particuliers et nous discutons à présent du singleton classique.

2 Le singleton basique

L'idée du singleton est simple. On écrit une classe dotée d'un constructeur privé pour éviter une instanciation extérieure et on donne un point d'accès global sous la forme d'une méthode statique retournant une instance de la classe, et instanciant la classe si besoin. Voici ce que cela donne :

```
public class SingletonBasic {

    private static SingletonBasic instance = new SingletonBasic();

    private SingletonBasic() {
    }

    public static SingletonBasic getSingletonBasic() {
        return instance;
    }

    public void doJob() {
        System.out.println("Je bosse comme un singleton");
    }
}
```

Très bien. Enfin apparemment, car quantité de problèmes potentiels nous guettent déjà. Commençons par le plus évident. L'instance est créée au chargement de la classe. Autrement dit, nous ne savons pas quand. Cela peut convenir, mais il est des cas où la création de notre objet demandera des informations qui ne sont pas encore disponibles à ce stade. Pour cette raison, on préfère une instanciation paresseuse c'est-à-dire à la demande. Notre classe devient :

```
public class SingletonBasicLazy {

    private static SingletonBasicLazy instance;

    private SingletonBasicLazy() {
    }

    public static SingletonBasicLazy getSingletonBasicLazy() {
        if(instance == null)
            instance = new SingletonBasicLazy();
        return instance;
    }

    public void doJob() {
        System.out.println("Je bosse comme un singleton");
    }
}
```

Pouvons-nous nous sentir satisfaits ? On lit souvent dans les ouvrages spécialisés qu'à défaut d'être valable en environnement multithread, notre singleton est valable en monothread. En fait, la vérité est plus nuancée.

3 Java le dire à mon père

Le lecteur n'aura pas manqué de remarquer que nous reprochons au singleton "tout static" d'interdire l'héritage alors que nous faisons de même en déclarant le constructeur privé. Tel est le Design Pattern standard. Il tombe sous le sens qu'on adaptera selon les besoins. On peut, par exemple, déclarer le constructeur protected. Une autre solution très courante est d'écrire une classe de base abstraite. On sous-classe ensuite pour définir différents singletons. Un bon exemple est Toolkit qui est une classe abstraite, avec sa méthode getDefaultToolkit implémentée. Cette dernière instanciera, selon la plate-forme hôte, la sous-classe appropriée, comme on peut le vérifier en consultant les sources dans la JDK.

4 Deux c'est trop, zéro c'est trop peu

Voyons maintenant pourquoi nos singletons ne sont pas valables, mais avant, simplifions-nous la vie. Éliminons les cas difficiles que sont les conteneurs d'EJBs et les conteneurs de Servlets. Éliminons aussi le cas de deux JVM en action. Est-ce réellement un problème d'avoir une instance dans 2 JVM distinctes ? Ça peut l'être avec RMI par exemple. On se retrouve alors avec deux instances en réseau. Éliminons donc tous ces cas, car nous avons beaucoup de grains de café à moudre avec une seule JVM.
Quelle JVM au fait ? Savez-vous qu'avec une JVM 1.x notre singleton passe à la trappe du ramasse-miettes ? En effet, avant Java 1.2, c'était le sort de toute classe auto-référencée, ce qu'est notre singleton. Nous voilà avec zéro instance en mémoire. Pour contourner cela, on doit veiller à conserver une référence sur l'objet singleton à un autre endroit du code, généralement la main de l'application. Cette facétie doit nous servir de leçon : écrire un singleton Java peut obliger à nous méfier de la JVM qui sera utilisée. Nous reverrons cela plus loin, plusieurs fois. Nul n'est à l'abri d'une étourderie. C'en serait une, d'écrire un singleton sous-classant une classe qui implémente la méthode clone :-) Sans doute le mieux est-il de verrouiller la question une fois pour toute, en ajoutant une méthode à notre singleton :

```
public final Object clone() throws CloneNotSupportedException {
        throw new CloneNotSupportedException();
}
```

Une autre étourderie serait de charger deux fois un singleton sérialisé. Pour interdire ce cas de figure, on ajoute une autre méthode à notre singleton :

```
private Object readResolve() throws ObjectStreamException {
        return instance; // instance supposée valide
}
```

Enfin, il ne faut pas oublier le cas où une application utilise plusieurs chargeurs de classes (ClassLoader). Si c'est le cas, nous aurons une instance du singleton par ClassLoader. Avertis du problème nous y remédions en remaniant la méthode getSingleton pour forcer l'utilisation d'un ClassLoader bien précis. Voici un exemple :

```
public static SingletonCL getSingletonCL() {
ClassLoader currentClassLoader =
SingletonCL.class.getClassLoader();
if (instance==null) {
        if (!currentClassLoader.toString().startsWith("sun.")) {
            try {
                ClassLoader parent = currentClassLoader.getParent();

                Class la_classe =
parent.loadClass(SingletonCL.class.getName());
                    instance = (SingletonCL)la_classe.newInstance();
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else {
            instance = new SingletonCL();
        }
    }
    return instance;
}
```

Il est entendu que ceci n'est qu'un exemple à adapter selon les besoins. Ici, nous faisons comme si nous avions deux ClassLoader, le premier dérivant du ClassLoader par défaut. On suppose que le nom qualifié de ce dernier commence par "sun." : (voir le paquetage sun.misc.Launcher des JVMs). Si ce n'est pas le cas, alors nous sommes dans le ClassLoader dérivé. Nous remontons alors au parent, le ClassLoader par défaut, et nous chargeons la classe avec celui-ci. Évidemment, ce code est dépendant de la JVM. Ainsi, il n'est pas garanti du tout que le nom du ClassLoader par défaut commence par "sun.". Enfin, nous pouvons rencontrer des problèmes subtils. Par exemple, notre classe ainsi chargée ne pourrait être transtypée vers une classe/interface chargée par un autre ClassLoader. Ce problème a sa solution, mais cela sort du cadre de cet article. Arrêtons-nous là. Maintenant notre classe est fiable en contexte monothread.

5 Les accès concurrents

La solution idéale dans ce contexte est notre SingletonBasic. En effet il est garanti que :

```
private static SingletonBasic instance = new SingletonBasic();
```

sera exécuté au chargement de la classe avant toute utilisation. La classe java.lang.runtime est écrite ainsi. Mais nous l'avons dit, cela n'est pas toujours possible. Regardons donc notre classe SingletonBasicLazy de plus près :

```
if(instance == null)
instance = new SingletonBasicLazy();
return instance;
```

comporte un bug potentiel. Si un thread vient de passer le test de l'instance et qu'il est préempté par un second thread entrant dans la méthode, ce dernier peut lire que instance est nulle et créer lui aussi une instance. Certes, les probabilités que cela se produise sont faibles, mais on est jamais trop prudent. Il paraît que c'est un bug de cet ordre qui a fait planter le robot qui devait explorer Mars, ce qui fait quand même cher le bug :-) Le remède est tout simple, il faut synchroniser la méthode :

```
public synchronized static SingletonBasicLazy getSingletonBasicLazy() {
if(instance == null)
        instance = new SingletonBasicLazy();
    return instance;
}
```

moyennant quoi on est tranquille. C'est de cette façon que java.awt.Toolkit est écrite.

6 Le mieux est l'ennemi du bien

Les choses se gâtent dès que l'on pense optimisation. En effet, une méthode synchronisée peut réduire les performances d'un facteur de 100. Si nous pensons que notre getSingleton (nous simplifions ici la notation) peut être invoquée incessamment depuis plusieurs threads, nous écrivons naturellement :

```
public static Singleton getSingleton(){
        if (instance==null)
        {
                synchronized (Singleton.class){
                        instance=new Singleton();
                }
        }
        return instance;
}
```

Afin de n'entrer dans une section synchronisée que lors de la création de l'instance. Voici une vraie fausse bonne idée qui ne fonctionne pas. Si un second thread préempte un premier thread au moment où celui-ci vient de tester instance, il conclura lui aussi que instance==null et deux instances seront créées. Qu'à cela ne tienne, appliquons la technique de la double vérification :

```
public static Singleton getInstance()
{
  if (instance == null)
  {
    synchronized(Singleton.class) {
      if (instance == null)
        instance = new Singleton();
    }
  }
  return instance;
}
```

Sommes-nous enfin tranquilles ? Non. Aussi incroyable que cela puisse paraître ce code ne fonctionne pas non plus.

7 Exécution dans le désordre

Les spécifications permettent au compilateur, Java ou JIT, d'arranger le déroulement des instructions, pourvu que la sémantique reste respectée. Ainsi, lorsque nous avons :

```
instance = new Singleton
```

instance peut être affectée par le résultat de l'allocation mémoire opérée par la JVM AVANT que le constructeur ne soit appelé. Ainsi, un thread qui préempterait au moment défavorable, acquerrait un objet incomplètement initialisé. Nous touchons là à une particularité de Java. En outre, nous voyons que nous sommes tributaires de l'environnement d'exécution. Par exemple, ce serait une erreur de désassembler le bytecode produit par le compilateur afin de vérifier si les instructions s'exécutent dans le bon ordre. Même si c'était le cas, il n'y a aucune garantie qu'un compilateur Just-In-Time ne mettra pas tout par terre. Le contraire est même plus que probable. Et comme on n'a aucun moyen de savoir quel JIT sera utilisé, mieux vaut abandonner cette double vérification.

8 Ce volatile est un drôle d'oiseau

On trouve encore, dans les livres et sur le Net, des exemples "améliorant" la double vérification en déclarant l'instance comme une variable volatile.

```
private volatile static Singleton instance;
```

À notre humble avis, cela relève plus de l'acharnement thérapeutique que de la bonne programmation. D'abord, rien ne garantit que cela fonctionne. En effet, le modèle mémoire de la machine virtuelle et la sémantique de volatile ont évolué avec le temps. Même si on peut espérer que cela fonctionne avec les toutes dernières JVM, on risque d'avoir de grosses surprises avec une JVM même relativement récente. Enfin, si l'on y réfléchit, cette solution n'en est pas une. Volatile a pour but d'éviter que la variable soit dans un cache propre au thread, un cache propre au processeur, ou que le code avant l'écriture de la variable ne soit désordonné. Autrement dit, le compilateur ajoute un traitement pour que l'on soit sûr que la vraie valeur de la variable soit visible partout. À l'instar d'une directive synchronized, ce traitement est coûteux en terme de performance, et là où l'on pensait optimiser, en fait, on ne gagne rien.

En guise de conclusion

Il est bien connu qu'en informatique "L'optimisation prématurée est la source de tous les maux". Quand on code un singleton, on ne sait a priori pas, ou mal si son point d'accès sera invoqué à tout bout de champ. Pour cette raison, votre serviteur pense que c'est une mauvaise idée de vouloir optimiser l'accès à une instance de singleton dans le singleton. Ce n'est rien d'autre qu'une optimisation prématurée. D'ailleurs, si on examine les classes Toolkit et Runtime déjà mentionnées, nous constatons que leurs auteurs ne se sont pas lancés dans l'optimisation. Si, un singleton existant, on devait écrire du code invoquant à répétition ses méthodes dans un thread, alors sans doute est il possible d'organiser ce code afin de faire un seul appel a une méthode getSingleton synchronisée, et une fois la référence obtenue, entrer dans la boucle du thread tout simplement.

Frédéric Mazué fmazue@programmez.com

Se familiariser avec GDI+

La nouvelle API graphique de Windows, GDI+, est riche en fonctionnalités avancées de dessin et de manipulation d'images. Nous allons nous familiariser avec sa programmation en C++.

GDI+ est une API graphique qui est apparue avec Windows XP et Windows Server 2003 afin de procurer des fonctionnalités de dessin avancées sur ces systèmes et à terme remplacer l'API graphique GDI des Windows précédents. GDI+ sert aussi de support aux classes graphiques de la plate-forme .Net, C# et Visual Basic ne devant sans doute pas être en reste devant l'API Java 2D :-) GDI+ n'est pas un jeu de fonctions C comme l'est l'ancienne GDI, mais un jeu de classes écrites en C++. Une connaissance de base de ce langage est donc requise pour la manier. En outre, travailler avec cette librairie au sein d'une application en code natif impose certaines règles et aussi de travailler avec Unicode, ainsi que de flirter un tout petit peu avec la technologie COM. Nous allons découvrir cela ensemble, puis nous verrons comment, avec GDI+, convertir une image du format PNG vers le format JPEG, relativement facilement.

1 Les outils

Pour dessiner avec GDI+, ce qu'il faut avant tout c'est... GDI+. Disponible sur Windows XP et Server 23, il est possible de l'installer sur Windows 98, ME et NT et 2000, mais pas sur Windows 95. Pour obtenir GDI+, rendez-vous à http://msdn.microsoft.com et faites une recherche pour "Redistribuable GDI+". Pour pourrez alors télécharger gratuitement la dll de GDI+, et comme le terme redistribuable l'indique, vous aurez le droit de l'utiliser et aussi de l'installer sur un vieux Windows devant accueillir votre application.
Ensuite, il vous faut un compilateur C++. N'importe lequel fera l'affaire, mais votre serviteur pense que c'est une excellente idée de se servir de Visual C++ 2005 Express Edition, outil gratuit jusqu'en novembre de cette année et téléchargeable à partir de http://msdn.microsoft.com/vstudio/express/visualc/. Par défaut, cet outil permet de programmer la plate-forme .Net en C++, mais il est possible de modifier ses fichiers de configuration afin de pouvoir travailler avec la Windows Platform SDK, autrement dit, pouvoir développer des applications en code natif. Les instructions sont d'ailleurs données par Microsoft sur la page de téléchargement de Visual C++.
Visual C++ ou autre, votre compilateur aura besoin de fichiers en-têtes récents et du fichier Gdiplus.lib. Ceux-ci ne seront pas disponibles dans un vieux Visual C++ 6.0 par exemple. Vous devrez alors télécharger la Windows Platform SDK depuis http://msdn.microsoft.com, puis faire pointer votre outil de développement sur les répertoires \lib et \include de la SDK. Sur le Cd-Rom du magazine vous trouverez sept programmes d'exemple écrits avec Visual Studio 2005. Le code part d'un squelette d'application Win32 ou console, généré par Visual Studio.

2 Pour se faire la main

Commençons par écrire et compiler une application de type Hello World. Nous voulons tracer une ligne en diagonale dans une fenêtre et écrire une ligne de texte. Que votre application soit générée par Visual Studio ou pas, vous devrez

inclure l'en-tête de Windows et celui de GDI+. Les classes de GDI+ sont groupées dans un espace de nom et vous devez en informer le compilateur. Enfin, votre éditeur de liens devra incorporer Gdiplus.lib, ce que nous obtenons sous Visual avec un pragma, parmi d'autres possibilités. Au final, un programme GDI+ en C++ commence donc ainsi :

```
#include "stdafx.h" // spécifique Microsoft
                    // et contient #include <windows.h>

#include <Gdiplus.h>

using namespace Gdiplus;

#pragma comment(lib, "gdiplus.lib")
```

3 Avez-vous mal à l'en-tête ?

Nous n'avons pas encore écrit de code et pourtant, si sous Visual Studio, vous compilez votre application, à ce stade, vous aurez le désagrément de vous faire insulter par votre compilateur qui émettra une bonne centaine de messages d'erreur abscons. La cause du problème se niche dans l'en-tête stdafx.h, dont on peut éviter la présence pour une application console, mais pas pour une application Win32. Cet en-tête contient une macro :

```
#define WIN32_LEAN_AND_MEAN
```

Quand le pré-processeur traite windows.h et que cette macro est définie, certains en-têtes rarement utilisés (Microsoft dixit) ne sont pas inclus. Parmi eux, des en-têtes COM dont a pourtant besoin GDI+. Comme on peut le voir dans gdiplusimaging.h, GDI+ s'appuie sur des interfaces COM :

```
MIDL_INTERFACE("025D1823-6C7D-447B-BBDB-A3CBC3DFA2FC")
IImageBytes : public IUnknown
{
        // etc
};
```

En l'occurrence, les messages sont émis par le compilateur car IUnknown n'est pas définie. Pour remédier au problème vous pouvez retirer la macro #define WIN32_LEAN_AND_MEAN de stdafx.h ou bien rajouter a posteriori l'inclusion de l'en-tête nécessaire comme ceci :

```
#include "stdafx.h"
#include <unknwn.h>
#include <Gdiplus.h>
```

4 Les sandwiches Microsoft

Écrivons maintenant du code GDI+ proprement dit. Nous étudions le programme d'exemple nommé Ligne. Nous voulons installer un gestionnaire d'événements pour peindre le contenu de la fenêtre. En programmation Windows, les travaux de dessin sont faits dans un sandwich dont la première tranche de pain initialise un contexte graphique et la deuxième tranche de pain libère ce contexte, selon ce modèle de code :

```
case WM_PAINT:
hdc = BeginPaint(hWnd, &ps);
OnPaint(hWnd, hdc); // code de dessin dans le sandwich.
EndPaint(hWnd, &ps);
break;
```

Ce principe reste vrai avec GDI+, au moins en ce qui concerne les évènements de dessin. En outre, nous devons mettre en place un deuxième sandwich dans le code, dont le rôle est d'initialiser GDI+. On peut mettre ce sandwich n'importe où a priori. Toutefois, le mettre dans le gestionnaire d'évènement WM_PAINT est peu pertinent en raison de la perte de performance et surtout en raison du fait des objets GDI+ qui pourraient se situer hors de cette portée et éprouver un malaise existentiel avec une GDI+ désinitialisée au sortir du gestionnaire. L'endroit classique pour traiter cela est tout simplement le début et la fin du main:

```
int WinMain
{
        // Initialisation de GDI+
        GdiplusStartupInput gdiplusStartupInput;
        ULONG_PTR           gdiplusToken;
        GdiplusStartup(&gdiplusToken, &gdiplusStartupInput, NULL);

        // du code ici

        // Fermeture de GDI+
GdiplusShutdown(gdiplusToken);

        return // etc
}
```

5 Premiers tracés

Toutes ces étapes préparatoires accomplies nous pouvons enfin tracer notre ligne et écrire notre texte.

```
VOID OnPaint(HWND hWnd, HDC hdc)
{
        char* texte = "Programmez!";
        WCHAR* buffer;
        int size;
        RECT rect;

        Graphics graphics(hdc);
        Pen      pen(Color(255, 0, 0, 255));

        ::GetClientRect(hWnd, &rect);
        graphics.DrawLine(&pen, 0, 0, rect.right , rect.bottom);

        SolidBrush  brush(Color(255, 0, 0, 255));
        FontFamily  fontFamily(L"Times New Roman");
        Font        font(&fontFamily, 24, FontStyleRegular, UnitPixel);

        size = strlen(texte)+1;
        buffer = new WCHAR[size];
        ::MultiByteToWideChar(CP_ACP, 0,
                texte, -1,
                buffer, size);
        PointF pointF((REAL)10, (REAL)(rect.bottom - 30));
        graphics.DrawString(buffer, -1, &font, pointF, &brush);
        delete[] buffer;
}
```

Commentons ce code. Nous commençons par construire un objet graphique autour du contexte graphique fourni par le système. Si nous nous trouvions dans un endroit du programme où le système ne fournit pas de contexte, la réaction à un clic sur un bouton par exemple, nous pourrions quand même travailler avec l'objet Graphics qui sera alors construit à partir d'un des autres constructeurs disponibles. Nous construisons ensuite un objet crayon (Pen). En GDI+, les API Windows ne perdent pas leurs droits et nous en employons une pour connaître les dimensions de la fenêtre. Et enfin nous traçons notre ligne avec la méthode DrawLine de Graphics. Notons que GDI+ est sans état. Ainsi, si nous voulions tracer une autre ligne bleue immédiatement après la première, nous devrions quand même passer l'objet Pen en argument à DrawLine. En cela, le travail avec GDI+ se démarque par rapport au travail avec les APIs WIN32.
L'instanciation de FontFamily nous fait découvrir que GDI+ travaille avec Unicode, ce qui est parfois contraignant. Quand le texte est connu à l'avance la macro L nous tire d'affaire. Sinon, on peut avoir besoin de faire une conversion d'Ansi vers Unicode via l'API MultiByteToWideChar, comme le montre l'exemple. Enfin, on remarque que les objets GDI+, instanciés sur la pile, seront détruits en quittant la portée. Il est évident que si les objets sont créés sur le tas par l'opérateur new, il faut les détruire explicitement avec delete.

6 Rendre une image

L'exemple suivant, AutomaticScaling montre comment rendre et croper une image *(voir ci-contre).* En voici le code intéressant :

```
VOID OnPaint(HWND hWnd, HDC hdc)
{
        #define CROPWIDTH 85

        Graphics graphics(hdc);

        Image image(L"Tux.png");
```

```
        // Redimensionnement automatique
        graphics.DrawImage(&image, 10, 10);
        // Pas de redimensionnement
        graphics.DrawImage(&image, 10, 210,
                image.GetWidth(), image.GetHeight());
        // cropage
        Rect rect(270, 210, CROPWIDTH, image.GetHeight());
        graphics.DrawImage(
                &image,
                rect,
                0, 0,
                CROPWIDTH,
                image.GetHeight(),
                UnitPixel);
}
```

L'image vient d'une capture effectuée sur Linux, plate-forme dont la résolution est de 72 pixels au pouce. Sous Windows, nous sommes en 96 pixels au pouce. Dans ce cas GDI+ redimensionne automatiquement l'image (image du haut sur l'illustration). Pour éviter cela, nous devons demander à l'objet image lui-même ses dimensions pour le tracer, comme le montre la seconde invocation de DrawImage, et l'image en bas à gauche sur l'illustration. Le reste du code concerne le cropage.

7 Manipuler les formats d'images

Le lecteur saura facilement manipuler les autres fonctionnalités de dessin de GDI+. Seul le maniement des formats d'images n'est pas très clair. Nous nous y attaquons maintenant. Les formats d'image, autrement dit les encodeurs/décodeurs connus du système, sont désignés pour l'utilisateur par des types Mime. Par exemple, image/png pour le format PNG. Il n'y a pas nécessairement un encodeur correspondant à un décodeur reconnu. L'exemple ListeEncodeurs liste tous les encodeurs et devrait afficher sur votre machine :

```
image/bmp
image/jpeg
image/gif
image/tiff
image/png
```

Tandis que les décodeurs seront probablement :

```
image/bmp
image/jpeg
image/gif
image/x-emf
image/x-wmf
image/tiff
image/png
image/x-icon
```

Obtenir les types Mime est simple. On procède de la même façon pour les encodeurs ou décodeurs.

On commence par un appel à GetImageEncodersSize pour obtenir la taille en octets d'un tableau devant recevoir des structures ImageCodecInfo, ainsi que le nombre d'éléments du tableau. Ensuite, un appel à GetImageEncoders remplit le tableau qu'il suffit de parcourir pour afficher les types Mime. Comme il se doit avec GDI+, les chaînes seront en Unicode. L'exemple ListeEncodeurs montre comment convertir de l'Unicode en caractères Ansi au moyen des APIs Windows, si on ne veut ou ne peut utiliser la librairie standard C++ pour la sortie sur console. L'exemple WListeEncodeurs procède lui en utilisant les flots de caractères étendus de la librairie standard C++.

Les deux derniers exemples convertissent une image png en trois images jpeg au taux de compression 0, 50 et 100 respectivement.

L'un travaille avec les APIs Windows pour Unicode et l'autre avec les flots de C++. Le principe est tout simple. On invoque la méthode Save d'un objet en lui passant des paramètres d'encodage. Le point délicat est que la méthode attend non pas un type Mime pour connaître le format, mais un GUID correspondant. Les GUID sont des nombres de 128 bits servant d'identificateurs uniques pour le recensement des interfaces COM. Nous avons vu que GDI+ travaille avec COM sous le manteau, ce qui justifie ici le besoin de passer un GUID en argument. (Quoique... Microsoft aurait peut-être pu fournir une méthode recevant le type Mime...) Bref, il faut obtenir ce GUID. On le trouve dans les structures ImageCodecInfo de manières similaires au type Mime.

Ces exemples doivent être étudiés avec soin car nous touchons là à COM et ses fuites de mémoire potentielle. L'API réserve de la mémoire 'COM' si l'on peut dire et le programmateur doit libérer cette mémoire au moyen de l'API CoTaskMemFree.

■ **Frédéric Mazué** fmazue@programmez.com

Méfiez-vous du Sudoku !

La folie du Sudoku a ses dangers : selon plusieurs sources, dont Panda Sotware, deux logiciels de Sudoku, l'un téléchargeable, l'autre directement jouable en ligne, ainsi que quelques sites pornographiques, installent un nouveau spyware sur votre PC, le bien dénommé YazzleSudoku. Rien de bien méchant, surtout pour la version téléchargeable, puisque le spyware prévient quand même (confusément) de son installation. Mais soyez prudent si vous ne voulez pas remplir votre écran avec des pop-ups au lieu de grilles avec des chiffres... **Site :** www.secunia.com/virus_information/28140/yazzlesudoku

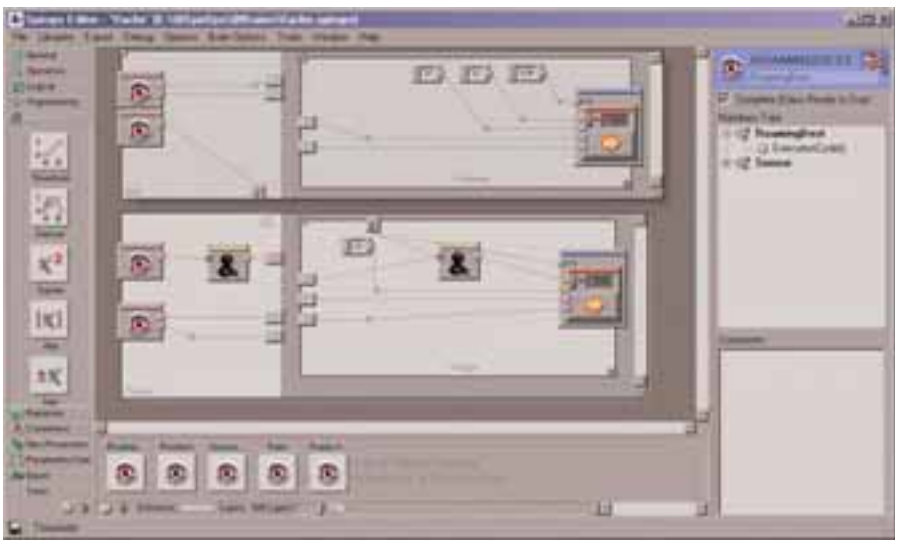
Vista-Center ?

En ce qui concerne le futur et toujours fluctuant nouveau système d'exploitation de Windows, Vista, on peut être au moins sûr d'une chose : il est prêt à dialoguer avec la Xbox 360 ! En fait, il la transforme en media-center, de quoi véhiculer, utiliser, diffuser et enregistrer sur ses équipements domestiques (TV, chaîne, etc.) tous les contenus multimédia de son PC. L'écran ci-dessous montre la pleine reconnaissance de la Xbox 360 depuis une version (non trafiquée !) développeur de Windows Vista. Alléchant, si Vista voit le jour, un jour...

Microsoft mange du lion

Microsoft vient de racheter Lionhead, le studio de développement du génial Peter Molyneux (Populous, PowerMonger, Magic Crapet, Black & White...). Même si les deux parties semblent

Coup de pouce... intelligent ?



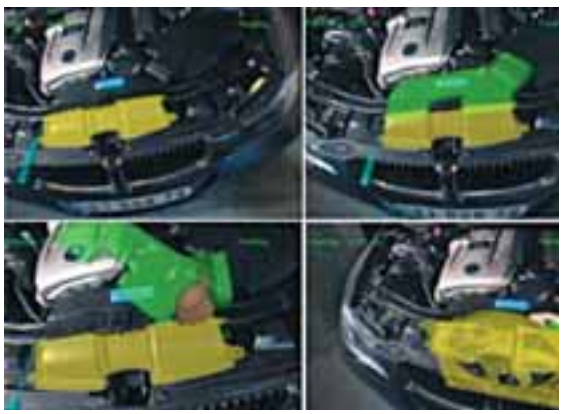
SpirOps AI est un middleware d'Intelligence Artificielle reconnu sur le marché des jeux, utilisé par Ubi Soft ou Quantic Dream entre autres. Une de ses caractéristiques les plus notables est son interface très graphique, qui permet de visualiser et de mettre concrètement en place les arbres décisionnels. Performant, mais très cher, et destiné aux pros... Alors la société SpirOps vient de décider de lancer un Starter Pack de son logiciel, c'est-à-dire une offre promotionnelle destinée aux développements indépendants en devenir : 5 000 € pour un nombre illimité de postes et de plates-formes, de quoi assurer sans trop se ruiner (!) une pré-production sérieuse pour un studio indépendant, et peut-être vendre ou finaliser le projet. Axel Buendia de SpirOps en fait une profession de foi : « Nous rencontrons tous les jours des équipes motivées, créatives mais possédant en général des ressources limitées en phase de prototypage. Nous souhaitons soutenir l'originalité de tous ces projets. Tous les développeurs peuvent désormais créer et valoriser l'IA de leur projet grâce à un outil de modélisation d'IA déjà utilisé dans des jeux AAA. » Bon, mais en cas de succès, il faudra ensuite acheter une licence Publishing pour SpirOps AI... La démarche n'est donc pas désintéressée ! Elle constitue cependant une innovation prometteuse, à laquelle on aimerait voir d'autres middlewares de développement ludique, souvent indispensables, se rallier... **Site :** www.spirops.com

Nintendo joue la Révolution

Le PDG de Nintendo appelle à l'innovation : dans sa toute dernière conférence de presse à la Game Developers Conference de San Jose en Californie, il a expressément demandé aux développeurs de sortir des sentiers battus et de chercher de nouvelles voies aux jeux vidéo. Et il semble que ce ne soit pas seulement un vœu pieux : Satoru Iwata s'est en même temps engagé à fournir aux développeurs, à égalité avec ses équipes internes, tous les outils nécessaires pour, selon son expression, « bouleverser les méthodes routinières de création de jeu ». Avec, en outre, un mode de téléchargement des jeux propre à l'innovation.

Le PDG Iwata faisait ainsi d'abord allusion à la manette de jeu

Total Immersion : une technologie hors norme !



Sensation durant la journée Microsoft DevDays de Paris, le 23 mars dernier. Une grande surprise attendait les participants. On pouvait voir de ses propres yeux une technologie 3D totalement révolutionnaire, et Française, combinant les prises de vue réelles et des objets 3D, capable d'interagir avec son environnement, le tout en temps réel ! France Télévisions avait utilisé cette technologie pour les fameux personnages / objets 3D que le présentateur montrait à l'écran durant les derniers jeux olympiques ! L'éditeur a mis au point son outil D'Fusion qui permet de réaliser cela. Cette technique s'appelle la réalité augmentée.

Le plus fort est que la technologie a été développée avec les technologies Microsoft : DirectX et DirectShow, le tout codé en C++ avec Visual Studio 2003. DirectShow sert à l'acquisition vidéo et DirectX permet de réaliser le rendu. Le principe de D'Fusion est simple. Il s'agit d'acquérir des scènes à partir de caméras, via une carte d'acquisition. C'est à partir de ce flux vidéo que l'outil injecte le flux 3D pour donner en sortie un flux de réalité augmentée. Quand on bouge, l'outil recalcule automatiquement les positions, les objets, via de puissants algorithmes temps réel. Cette technologie rencontre déjà un beau succès auprès d'industriels, notamment dans l'automobile. Le flux 3D, les objets 3D sont créés et manipulés à partir d'un langage script créé par Total Immersion. Il n'existe pas encore un véritable IDE « convivial » mais cela fait partie des projets de la société. L'outil fonctionne d'ores et déjà en mode haute définition.

Il est possible que dans les prochains mois, l'éditeur passe sous Visual Studio 2005. Pour le moment, rien n'est décidé sur l'utilisation du langage C# à la place du C++... mais la porte n'est pas fermée !

Site : www.t-immersion.com

■ **F.T.**

« révolutionnaire » de la Révolution, cette sorte de télécommande-joypad dont les mouvements dans l'espace peuvent se reproduire à l'écran. De quoi exciter l'imagination des développeurs pour mettre en place et gérer de nouvelles sensations de jeu.

Mais l'incitation-massue de Nintendo au développement indépendant et original, c'est surtout son service Virtual Console, intégré à la Révolution, qui permet de télécharger des jeux. Donc, pas de distribution en magasins, pas de cartouches ni d'hyper-DVD propriétaires, bref, une distribution purement numérique qui devrait permettre à toutes les créations audacieuses, et même réduites, de tenter leur chance.

Plein accès aux outils, nouveau périphérique de commande, accès numérique « low-cost » : pour une fois, il semble que Nintendo joue aussi la carte des développeurs indépendants ! A suivre, dans la pratique...

se féliciter de leur union, il reste à savoir si Molyneux pourra bénéficier de la même liberté de création, et surtout des mêmes incroyables retards de développement dont il était le spécialiste. On peut cependant voir là une volonté de Microsoft d'offrir des titres plus adultes, fouillés, et ludiquement innovants à sa Xbox... Heureuse nouvelle dans ce cas !

Tuning sur PS3

Le développement sur PlayStation 3 prend meilleure tournure : SN Systems, société anglaise qui appartient à Sony, spécialisée dans les outils pour consoles, vient d'annoncer la disponibilité de son utilitaire Tuner sur PS3. Tuner, déjà connu sur PS2 et PS1, sert à la détection des conflits ou des goulots d'étranglement dans le code-source des jeux PlayStation. Evidemment, la structure particulière de la PS3 a nécessité une refonte presque complète de Tuner, dont on retrouve cependant similaires l'interface et l'organisation des fonctions. L'outil est accessible à l'essai et à la vente pour les développeurs enregistrés. **Site :** www.snsys.com

Développez ludique mais "serious"...

L'Apply Serious Games 2006 se tiendra à Londres les 25 et 26 Mai prochains. Mais que sont les « Serious Games » ? Tout simplement les programmes de jeux destinés aux institutions, éducation, armée, association, etc., dont le marché va croissant (28% par an depuis 2003). Une voie intéressante de développement, propre aux innovations, vers laquelle se tournent, en partie ou complètement, des studios de plus en plus nombreux. La filière institutionnelle, très demandeuse de Serious Games, présente aussi l'avantage de lancer des appels d'offres et donc, en cas de sélection, de garantir des financements solides... A découvrir et explorer dans la capitale british : www.applyseriousgames.com/pages/expo.htm.

SQL Server 2005

- *Auteur : William R. Stanek*
- *Editeur : Microsoft Press*
- *Prix : 39 €*

Avec SQL Server 2005, l'administrateur doit acquérir de nouveaux réflexes, de nouvelles habitudes de travail. Ce petit pavé décortique l'ensemble des modules et fonctions liés, de près ou de loin, à l'administration au quotidien du serveur, des bases, des données. Il y a beaucoup de nouveautés, notamment dans la sécurité et les différents modes d'autorisation d'accès qu'il faut comprendre et maîtriser. L'ensemble des problématiques d'administration sont évoqués : déploiement, sécurité, configuration, import / export, sauvegarde, etc. Un petit focus sur la version Express aurait été un plus intéressant pour le lecteur. L'ouvrage est clair et contient de nombreux exemples pour mieux comprendre. Bref, un ouvrage de base pour tout administrateur SQL Server.

Collection Et maintenant Programmez ! (Microsoft Press)

- Visual Basic 2005 Express Edition (19,90€) : comment démarrer sur un outil et un langage inconnu, même si l'IDE choisi se veut simple d'accès ? Pour tenter de maîtriser la bête, rien ne vaut un exemple d'application de bout en bout pour aborder les principaux aspects. C'est le principe de cette collection Microsoft Press. Chaque chapitre est construit sur un aspect très visuel et didactique. À la fin de chaque partie, le résumé des pages précédentes. Si le code est présent, on privilégie les manipulations visuelles. L'ouvrage ressemble plutôt à un « démarrer avec » car il est difficile de démarrer la programmation avec cet ouvrage, faute d'un guide de référence du langage. Il est dommage qu'aucun CD n'accompagne l'ouvrage, ç'aurait été un plus pour démarrer encore plus rapidement !
- Autre titre disponible : Visual C# 2005 Express Edition

Développement web avec J2EE

- *Auteur : Éric Sarrion*
- *Editeur : O'Reilly*
- *Prix : 35 €*

Comment développer efficacement des applications Web avec J2EE ? C'est tout le défi d'aller à l'essentiel dans un livre concentré en 240 pages ! L'auteur pose tout d'abord les outils utilisés : Jrun, Tomcat, Eclipse, Lomboz. Puis s'ensuivent les éléments fondamentaux pour J2EE et le web : servlet, JDBC, JSP, le tout à la mode MVC. Le contenu est très didactique avec des explications claires, sans informations inutiles. L'auteur mise beaucoup sur l'exemple, le code. Il s'agit d'une bonne introduction au développement Web avec Java / J2EE. Seul le prix paraît un peu élevé.

De l'art de programmer en Perl

- *Auteur : Damian Conway*
- *Editeur : O'Reilly*
- *Prix : 43 €*

Perl demeure un langage incontournable, mais savez-vous réellement bien coder en Perl et utiliser les bonnes pratiques ? Cet ouvrage est là pour réaliser un code clair, maintenable et en utilisant les meilleures méthodes de codage. L'auteur nous gratifie de plus de 250 recommandations et aborde tous les aspects : nommage, présentation du code, structure, rédiger une documentation, améliorer les boucles et diverses autres opérations, etc. Chaque recommandation est classée dans une catégorie, puis on pose le problème et son cadre d'utilisation, ensuite on explique, avec exemples à l'appui. Si l'ouvrage est dédié à Perl, les autres développeurs pourront trouver quelques idées pour leur langage. Un livre indispensable !

120 scripts pour Flash 8

- *Auteur : David Tardiveau*
- *Editeur : Eyrolles*
- *Prix : 29,90 €*

Flash constitue un formidable outil de développement Web et son langage, ActionScript, une mine de possibilités. Quand on développe, il est parfois utile de connaître des exemples de code, mais aussi d'avoir des codes prêts à l'emploi pour aller plus vite, ainsi que des astuces. Les scripts sont regroupés en plusieurs catégories et couvrent un spectre fonctionnel très large. Chaque script est présenté et le code expliqué pour que le développeur le maîtrise mieux et plus rapidement. Le niveau du player requis est indiqué à chaque script. Il aurait été intéressant de faire un focus uniquement sur les nouveautés Flash 8 et la mobilité. Les scripts sont téléchargeables sur le web.

Windows Vista

- *Auteur : Benoît Lanlard*
- *Editeur : Éditions ENI*
- *Prix : 27,14 €*

Alors que Windows Vista sortira fin 2006, début 2007, voici sans aucun doute le premier livre sur le prochain système de Microsoft. Il s'agit avant tout d'un guide pratique d'utilisation. Il aborde l'ensemble des tâches quotidiennes sous Vista, de l'installation à la sécurité. Les principales nouveautés sont présentées. Cependant, au niveau interface, il manque une vraie présentation d'AERO et sa mise en œuvre, idem pour Internet Explorer 7, bien peu présent. Les amateurs de petits gadgets seront ravis d'utiliser le clone du Dashboard d'Apple : Gabget. L'auteur s'attarde assez longuement sur les fonctions réseaux, et gestion des utilisateurs. Une bonne introduction à Vista en attendant une 2e édition plus complète au moment de la disponibilité du système.

HORS-SÉRIE

Programmez !

LE MAGAZINE DU DÉVELOPPEMENT

100 % .NET

Maîtrisez .Net et ses outils !

VISUAL BASIC • C# • VISUAL STUDIO • ASP.NET • SQL SERVER

Hors-Série - Avril/Mai 2006 • 5,95 €

C# / VB

Savoir utiliser le code snippet

Utiliser le déploiement ClickOnce

Gérer les captures d'écrans

WEB

Maîtriser le cache ASP.NET

Localiser les sites web

Simplifier le développement web avec la master page !

OUTILS

La pagination avec SQL Server 2005

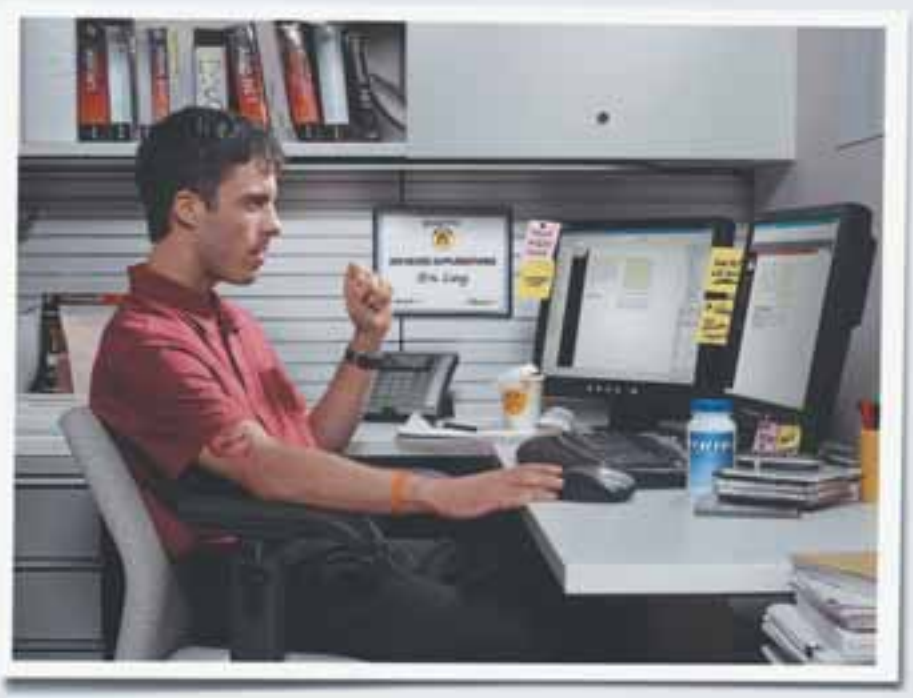
Expression : le nouveau développement web arrive !

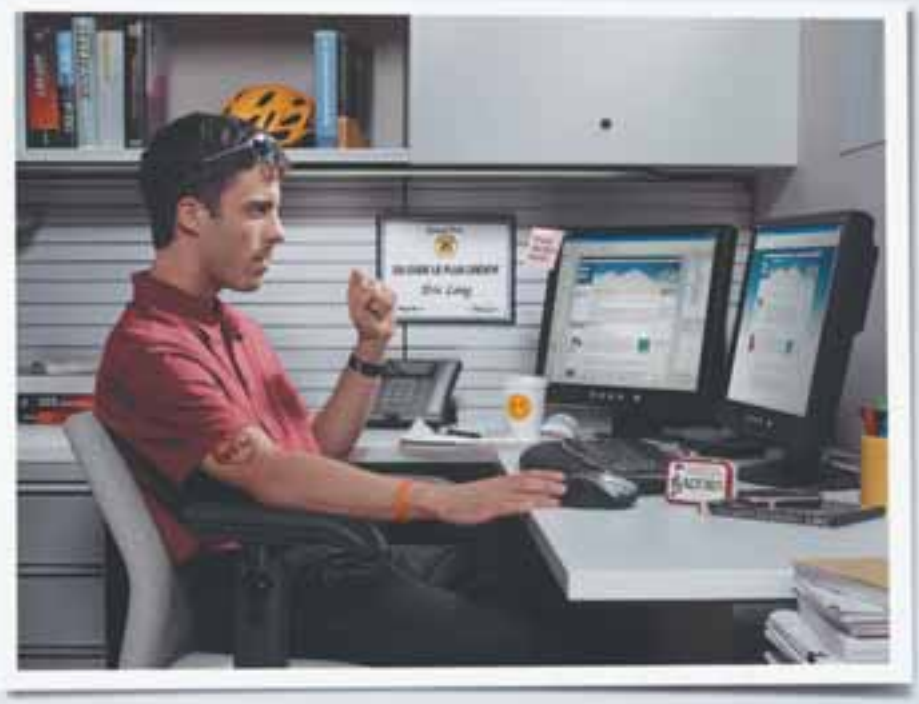
100 % PRATIQUE

En vente chez votre marchand de journaux

Votre potentiel, notre passion.™

Microsoft





Nouveau Visual Studio 2005. La différence saute aux yeux.

Vous voyez la différence ? Vous vous en rendrez compte dès la première ligne de code. Visual Studio 2005 propose plus de 400 nouvelles fonctionnalités pour vous aider à développer vos applications. Il vous permet d'accélérer vos développements en tapant moins de code, en déboguant plus efficacement, et en profitant des outils de tests avancés. Vous pouvez ainsi vous concentrer sur l'essentiel. Pour découvrir maintenant le nouveau Visual Studio 2005, consultez : **www.microsoft.com/france/vs2005**

Microsoft® Visual Studio® 2005