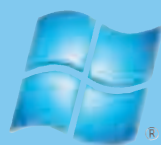


Gaming**NOUVELLE
FORMULE**

Développer pour Xbox Kinect



Quand les étudiants innovent !



Windows® Azure™

Le cloud pour tous les langages

🔥 WINDOWS

Rejouer
les actions
utilisateur

🔥 WEBMASTER

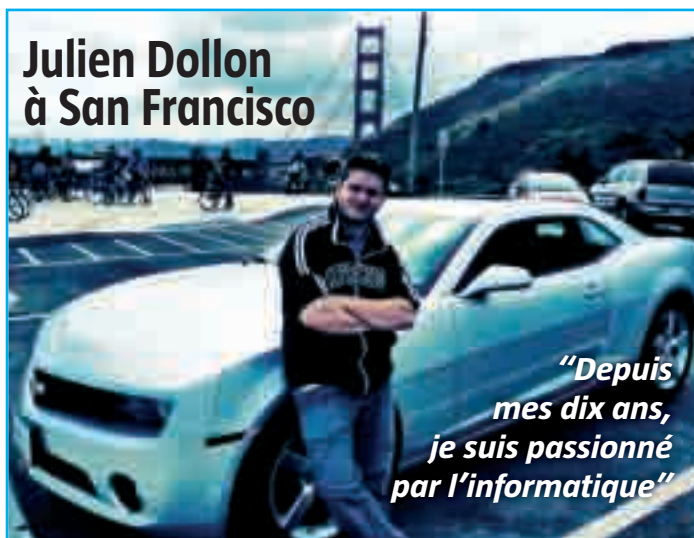
Utiliser les API
Géoportail

🔥 BUGTRACK

Le bug
PhotoChooserTask

Développeur du mois

**Julien Dollon
à San Francisco**



*"Depuis
mes dix ans,
je suis passionné
par l'informatique"*

CARRIÈRE

Comment se former à l'open source ?

M 04319 - 148 - F: 5,95 €



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

DÉVELOPPEZ 10 FOIS PLUS VITE

WINDEV®

- Windows 32 & 64 bits
- Linux
- Mac
- Internet
- Intranet
- Windows Mobile & CE
- Windows Phone
- Android 
- et maintenant
- iPhone et iPad.

- Développez vos applications une fois pour toutes.
Votre code, vos fenêtres, vos données sont compatibles.
Déployez vos applications sur tous les systèmes et tous les matériels.
Développez des applications portables, c'est facile avec WINDEV 17.
Vous aussi, développez 10 fois plus vite, pour toutes les plateformes.

**VERSION
EXPRESS
GRATUITE**
Téléchargez-la !

Intégralement en français.
Support Technique inclus.
Ouvert à tous les standards,
à toutes les bases de données

Est « Langage le plus
productif du marché »

WINDEV, WEBDEV
et WINDEV Mobile
sont compatibles



Nouveau:
créez des applications
iOS (iPhone, iPad)

**EXIGEZ WINDEV 17
POUR LE DÉVELOPPEMENT
DE VOS APPLICATIONS**

Fournisseur Officiel de la
Préparation Olympique



► **DEMANDEZ VOTRE DOSSIER GRATUIT**

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

www.pcsoft.fr

LeWeb11 : au-delà du réseau social

Durant trois jours les 7,8 et 9 décembre, le monde du web était présent, près de Paris, pour le traditionnel «LeWeb».

Cette édition avait comme invité d'honneur Eric Schmidt, très en forme sur le sujet Google : il prédit que Android dépassera en nombre de développeurs iOS dans les prochains mois et que Google TV s'imposera dans les téléviseurs d'ici l'été 2012 (l'ami Eric a sans doute été pris par l'ambiance). Au total plus de 3 000 personnes se seront bousculées dans les allées (parfois étroites) des trois halls réservés à l'événement. Mais LeWeb demeure aussi le lieu où il faut être, se faire voir, et accessoirement montrer sa vision (mot magique aussi imprécis que passionnant).

Loïc LeMeur est toujours à la barre de l'événement et le principal thème était le SoLoMo, ou social – local – mobile. Donc pas étonnant si le social était partout. Mais à y regarder de plus près, LeWeb11 était intrinsèquement plus intéressant que les années précédentes particulièrement sur la partie startup et les technologies présentes, bien loin du thème unique et imposé du SoLoMo.

Les réseaux sociaux étaient omniprésents sur de nombreux stands, dans les sessions, dans les rencontres : IBM, Google, Salesforce, Orange pour ne citer qu'eux. Le SoLoMo peut se décrire ainsi : le réseau social, les données sociales localement, avec une dominante géolocalisée et de précision, avec la nécessité d'être de plus en plus mobile, c'est-à-dire accessible de partout sur n'importe quel terminal. La tablette et le smartphone ont été les vedettes incontestables de l'événement. Mais le problème soulevé par le SoLoMo est important : quelle stratégie globale ? Comment le mettre en œuvre ? Quel respect des données privées ? Pour cela, il fallait rencontrer les éditeurs et les startup pour prendre conscience des difficultés réelles et sortir du simple discours. Et là, on découvre tous les problèmes, par exemple, comment mesurer la notoriété, l'impact d'une marque, d'une informatique sur un réseau social ? Tequilarapido a développé un service dédié à cette question sensible par



exemple (reposant sur Windows Azure). IBM expliquait sa vision du réseau social en entreprise. Salesforce exposait son tout nouveau Radian6.

DU CLOUD, DES APP STORE

Internet reste le cœur du LeWeb mais constatons tout de même que plusieurs technologies et pratiques ont fait de l'ombre au tout web de l'événement. Ainsi, Intel était largement présent avec son App Store : AppUp. Le but est clair : montrer la bou-

tique Intel, promouvoir les développeurs et renforcer la partie logicielle du fondeur qui a toujours du mal à se faire un nom. AppUp est pourtant une belle initiative avec de bonnes technologies (dont l'incontournable HTML 5). Salesforce était là aussi pour son PaaS open source : Heroku dont la popularité n'est pas à faire. Aujourd'hui, ce PaaS mise sur Java, PHP, Ruby pour asseoir sa notoriété. Et les déploiements explosent depuis un an : plus de 36 000 en octobre 2011 contre 18 000 un an auparavant. Et dans la foulée de cette bonne santé, un nouveau patron a été nommé, Byron Sebastian, qui possède une solide expérience des plateformes commerciales.

LES NOUVELLES EXPÉRIENCES D'INTERFACE

Au-delà du web, les nouvelles interfaces étaient bien représentées, que ce soit les nouveaux écrans interactifs de type « magic mirror », Kinect ou encore la réalité augmentée. Un des projets les plus intéressants était e-kinox de e-figy. Il s'agit d'une borne Kinect avec tablette et caméra, une expérience utilisateur surprenante, notamment avec le « magic mirror ». Il s'agit d'un concept assez impressionnant par son approche à la fois ludique et technologique. On peut créer une cabine d'essayage virtuelle grâce à la technologie Kinect de Microsoft mais l'interface passe par une

« tablette » pour affiner le choix des produits. Autre projet orienté Kinect et réalité augmentée, les idées de 3dvia. Il s'agit d'une application de conception de 3D immersive dont l'utilisation passe par la console Kinect. C'est tout bonnement bluffant. Un exemple parmi d'autres : <http://www.youtube.com/watch?v=iC07QpQ8J8Q>

Ces technologies démontrent parfaitement comment ces interfaces naturelles peuvent réellement faire évoluer notre contact avec une application, un usage. Avouons que ces expériences nous ont séduits !



11

BUZZWORLD

LeWeb11 : au-delà du réseau social3

ACTUS

En bref6

OUTILS

Explorez votre code Java avec JavaDepend.....12

MATÉRIEL

.....14

SÉCURITÉ

Les failles du Web :
petite analyse d'un site web18

VEILLE TECHNO

La FOLIE DES PROJETS D'ÉTUDES DE L'EPITECH !

Le retour du développement natif en C++21

RESSOURCES

Notre sélection de livres28

Agenda29



21

Le DÉVELOPPEUR DU MOIS

Julien Dollon.....30



Je DÉBUTE...

J'installe Android SDK 4.032

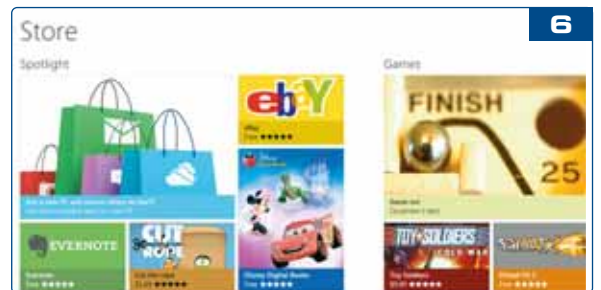
Installation de CUDA C avec Visual Studio 201054

CARRIÈRE

Profils d'étudiants Open Source.....51

Un stage pour un futur jeune diplômé52

Tendances de l'emploi informatique.....53



6

À la découverte de Windows Azure

• Essayez Windows Azure !3

• Ma première application Windows Azure.....4

• Windows Azure : une plateforme
pour tous les développeurs6

• Les données avec SQL Azure
et Windows Azure Storage8

• La synchronisation de la donnée
en mode Cloud.....10

• Windows Azure pour les sites Web12

• Windows Azure et la mobilité14



A lire dans le prochain numéro

Février 2012 - n°149, parution le 31 janvier 2012

♦ **Windows 8 :**
Au coeur de WinRT

♦ **Webmaster**
Traquer les bugs
de son site web

♦ **Coding4Fun**
Prendre le contrôle de son
PC avec son smartphone

♦ **Cahier spécial :**
Qualité logicielle avec
les outils Microsoft



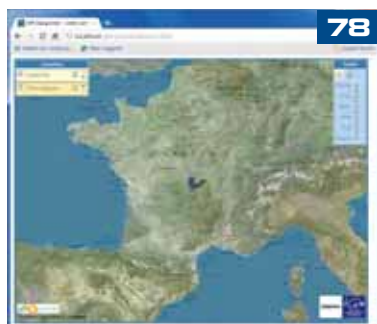
14

CAS D'USAGE

Accéder aux données SharePoint
avec REST & JSON.....58

PRATIQUE

Applications
bureau riches
en Silverlight
avec le mode
Out-Of-
Browser.....61



78

Rich Ajax Platform :
le meilleur d'Eclipse pour le Web.....66

Réjouer les actions des utilisateurs
sous Windows70

GAMING

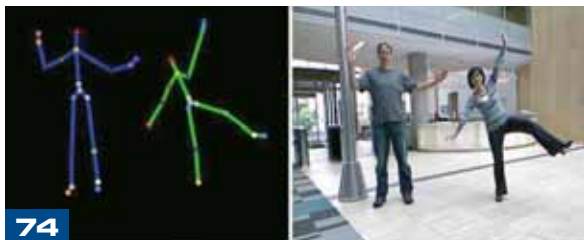
KINECT SDK :
éclate-toi !.....74

WEBMASTER

Mettre en oeuvre l'API Géoportail.....78

BUGTRACK

Le bug
PhotoChooserTask.....82



74



Programmez ! c'est vous !

Comme vous le constaterez au fil des pages, Programmez ! change profondément dans la forme et sur le fond. Le magazine existe depuis 13 ans et il doit évoluer avec son temps, avec les nouvelles habitudes du lecteur : vos habitudes.

Nous avons voulu par cette refonte totale replacer le développeur au cœur de l'action : vous montrer au quotidien, parler technique et programmation mais aussi de la personne, du métier, des évolutions de celui-ci. Ces dernières années, nous n'avons sans doute pas mis assez en avant votre job, votre quotidien, en oubliant que derrière la technologie, il y a une femme, un homme. La rubrique « carrière et emploi » abordera dans chaque numéro les métiers, le marché de l'emploi, les formations, car le métier de développeur évolue constamment et il vous faut des informations précises et nouvelles sur votre carrière.

La technique demeure au cœur du magazine ! Que serait Programmez ! sans lignes de code ? Nous nous sommes longuement interrogés sur la rubrique « code », comment l'améliorer, mieux l'utiliser. Si la pure technique reste, nous avons voulu mieux mettre en avant le côté pratique, « vraie vie », de la technologie : comment faire ? Comment démarrer avec telle technologie, comment intégrer telle librairie dans un vrai projet. Ce sont les ambitions des rubriques « je débute avec », « pratique », « webmaster ». Mais qui dit code, dit forcément bug. La rubrique « bugtrack » exposera chaque mois un bug et proposera une ou plusieurs solutions pour le résoudre.

Pour coller au plus près de l'actualité, trois rubriques décortiqueront les principales nouveautés, les tendances, pour comprendre le marché, les évolutions : buzzworld, veille technologique et actualités. Mais nous n'oublions pas que dans tout développeur se cache une âme de geek, les pages « matériel » parleront des nouveaux matériels / gadgets mais aussi des matériels pour mieux travailler au quotidien.

Comme vous le constaterez la page « livres » disparaît au profit des pages « ressources ». Ces pages vous donneront chaque mois un choix des livres récents, un agenda des prochains événements mais aussi toute l'actualité des communautés, une sélection de sites et de blogs à ne pas manquer.

Pour vous faciliter le lien entre le magazine et Internet, nous généraliserons durant les prochains mois l'usage du flashcode : une capture du tag et la page web encodée sera automatiquement affichée sur votre smartphone !

Plus que jamais, nous avons besoin de vous. Sans lecteurs, sans communauté, Programmez ! n'existerait pas. Venez témoigner, décrire un bug vicieux, expliquer une intégration technique. N'hésitez pas à nous contacter : redaction@programmez.com

François Tonic
Rédacteur en chef

*Toute l'équipe de PROGRAMMEZ !
vous souhaite
une excellente année 2012.*

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

Rédaction : redaction@programmez.com

Directeur de la Rédaction : Jean Kaminsky. Rédacteur en
Chef : François Tonic - ftonic@programmez.com. Ont colla-
boré à ce numéro : F. Mazué, S. Saurel, Y. Grandmontagne
Experts : I. Lahiali, C. Pichaud, B. Boucard, M. Bidon,
A. Zemzemi, J. Thiriet, N. Calvi, S. Blanchard.

Illustrations couverture : ©Microsoft, D.R.

Publicité : Régie publicitaire, K-Now sarl.

Pour la publicité uniquement : Tél. : 01 41 77 16 03 -
diff@programmez.com. Dépôt légal : à parution -
Commission paritaire : 0712K78366 ISSN : 1627-0908.
Imprimeur : S.A. Corelio Nevada Printing, 30 allée de la
recherche, 1070 Bruxelles Belgique.

Directeur de la publication : J-C Vaudecrane

Abonnement : Programmez, 17, Chemin des Boulangers, 78926
Yvelines Cedex 9 - Tél. : 01 55 56 70 55 - abonnements.programmez@groupe-gli.com - Fax : 01 55 56 70 91 - du lundi au jeudi de 9h30 à 12h30
et de 13h30 à 17h00, le vendredi de 9h00 à 12h00 et de 14h00 à
16h30. Tarifs abonnement (magazine seul) : 1 an - 11 numéros France
métropolitaine : 49 € - Etudiant : 39 € - CEE et Suisse : 55,82 € -
Algérie, Maroc, Tunisie : 59,89 € Canada : 68,36 € - Tom : 83,65 € -
Dom : 66,82 € - Autres pays : nous consulter. PDF : 30 € (Monde
Entier) souscription exclusivement sur www.programmez.com



» Ubuntu

perdrat de sa superbe ! Selon les statistiques de DistroWatch, Ubuntu subirait

une chute d'utilisateurs.

La distribution serait sur les 12 derniers mois 2e derrière Mint et pour le dernier mois (novembre 2011), elle n'arriverait que 4e derrière Mint, Fedora et OpenSuSe. Une des raisons pour expliquer cette chute, le changement d'interface par défaut : Unity. Mint connaîtrait la plus forte progression en un an.

» La fondation Linux

travaille à la version 5 de Linux Standard Base (LSB). La v4.1 est sortie en mars dernier. Le principal travail concerne la modularité de LSB pour mieux répondre aux ISV, développeurs. La notion de runtimes profiles a été évoquée. Cette v5 incorporera de nombreuses nouveautés (si le développement avance et que les objectifs sont respectés) : amélioration et correction de bugs du App Checker, support des derniers GCC, suppression de Qt3, support de nouveaux langages...

Pour en savoir plus :

<https://wiki.linuxfoundation.org/en/ProjectPlan42>

data.gouv.fr BETA
INNOVATION TRANSPARENCE . OUVERTURE

» Le gouvernement

français se met aux « open data ».

L'annonce officielle précise « En rendant accessibles les données publiques, l'Etat et les collectivités locales favorisent la création de nouveaux services et applications citoyennes, et participe au développement de l'économie numérique. Il renforce aussi la transparence de ses actions et promeut une meilleure information des citoyens. ». Il n'y a plus qu'à attendre l'exploitation de ces données publiques... site :

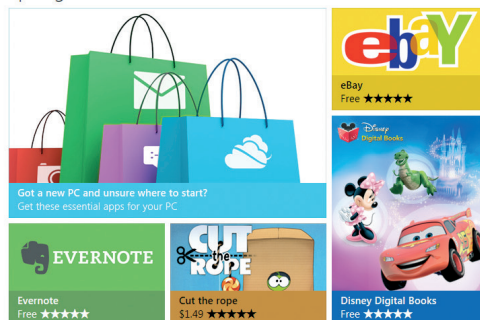
<http://www.data.gouv.fr/>

Windows 8

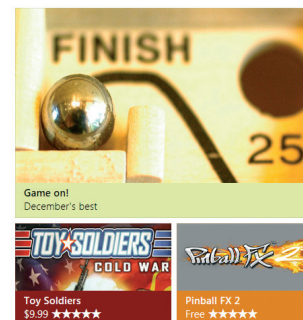
Les précisions se succèdent

Store

Spotlight



Games



A lors que la première bêta publique n'est pas attendue avant février 2012, les équipes de Windows 8 en dévoilent chaque semaine un peu plus sur le futur système. Outre la partie stockage (voir rubrique matériel), Win8 se dote d'une boutique applicative comme App Store d'OS X d'Apple. Windows Store sera une application Metro (nouvelle interface) en plein écran, utilisant massivement HTML 5 et JavaScript. Cette boutique sera mondiale et supportera 100 langues. Les arguments de Microsoft se veulent percutants :

- **De meilleurs revenus** : Avec le Windows Store, les applications les plus populaires aideront les développeurs à faire de meilleurs profits. Toutes les applications démarreront sur le standard du marché, soit une répartition de 70%/30%. Toutefois, pour les applications générant 25 000 dollars et plus, les développeurs pourront conserver 80% des recettes générées tout au long de la vie de leur application. Ce seuil de 25 000 dollars est international et s'applique à toutes les ventes quel que soit le pays en convertissant le montant de la devise.

- **Des conditions transparentes** : Le Windows Store offre un processus transparent, du chargement au téléchargement. Avec d'autres plateformes, le processus d'approbation des applications reste flou. Avec le Windows Store, il est non seulement possible de réaliser soi-même les tests de validation Microsoft avant de soumettre son application, mais également de suivre son évolution durant tout son cycle de vie.

- **Des modèles économiques flexibles** : Les développeurs ont la liberté d'utiliser le service de transaction de Microsoft ou le leur pour tous les achats « in-app », sans que Microsoft ne prenne de pourcentage sur les ventes.

- **Une visibilité structurée** : Une des priorités du Windows Store est d'assurer la promotion et la découverte des applications, notamment à travers des pages web descriptives indexées par le moteur de recherche Bing et Internet Explorer 10, ainsi que des périodes d'essai pour tester l'application.

La partie « setup » système sera en grande partie refondue notamment pour tout ce qui est processus d'installation afin d'éviter les multiples redémarrages. La diversité des plateformes matérielles pose un évident problème de compatibilité et de stabilité. L'installation devra déterminer la compatibilité matérielle et logicielle de la machine. L'installation en ligne, après téléchargement, sera possible pour faciliter la mise à jour. Sur le multi boot, l'utilisateur pourra passer en manuel par l'interface « advanced setup ». Exemple fourni par l'équipe de Windows 8 :

```
<ImageInstall>
  <OSImage>
    <InstallFrom>
      <Path> Z:\sources\install.wim </Path>
    </InstallFrom>
    <InstallTo>
      <DiskID>0</DiskID>
      <PartitionID>1</PartitionID>
    </InstallTo>
    <WillShowUI>OnError</WillShowUI>
    <InstallToAvailablePartition>false</InstallToAvailablePartition>
  </OSImage>
</ImageInstall>
```

Pour en savoir plus :

<http://blogs.msdn.com/b/b8/archive/2011/11/21/improving-the-setup-experience.aspx>

Microsoft®
tech·days
7, 8 et 9 février 2012

**SOYEZ AU CŒUR DU
PLUS GRAND ÉVÉNEMENT
INFORMATIQUE EUROPÉEN !**

➔ **6^{ème} édition**
Palais des Congrès Paris

- Pour les professionnels de l'Informatique.
- Pour les décideurs IT.
- Pour les développeurs.

AU PROGRAMME :

Participez à des sessions sur les enjeux actuels de l'informatique, pour tous les métiers : **Cloud Computing**, Intégration, **Technologies de développement**, Virtualisation, **Mobilité**, Sécurité, **Management, Décisionnel**,...

3 conférences plénières
300 sessions techniques
17000 visiteurs
140 exposants
50 communautés
35 000 web-spectateurs
TechDays TV



Inscrivez-vous
et suivez-nous
sur **www.mstechdays.fr**

En partenariat avec :

Le magazine du développement
PROgrammez !

Nvidia ouvre un peu CUDA : bonne ou mauvaise nouvelle ?

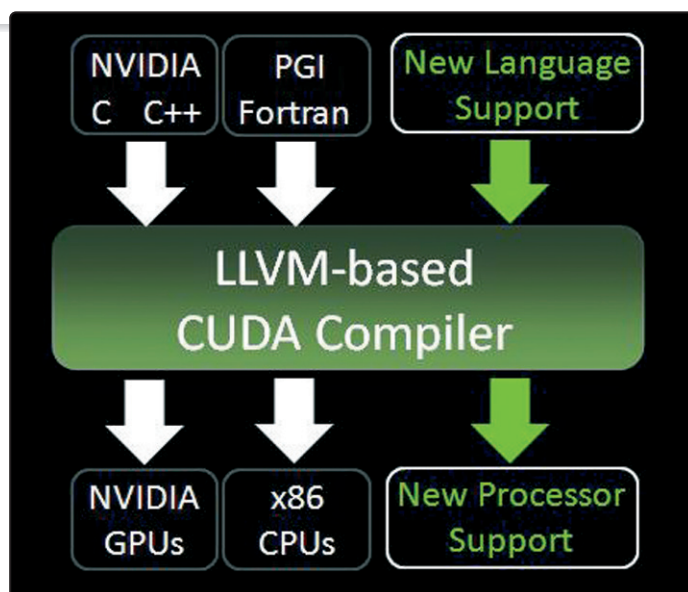
Nvidia a annoncé l'ouverture de sa plate-forme CUDA pour l'accélération matérielle et l'utilisation des GPU pour les applications. CUDA était coincé entre OpenCL et la nouvelle arrivée de Microsoft, sans oublier Intel. CUDA avait tout de même un peu de mal à se faire une place sur ce marché finalement très concurrentiel. L'annonce concerne la fourniture du code source du compilateur CUDA basé sur LLVM. Ce code sera accessible aux chercheurs, éditeurs, développeurs.

Cela va leur permettre d'ajouter plus facilement la compatibilité GPU à plus de langages de programmation et à plus d'applications compatibles CUDA, et ce, sur des architectures de pro-

cesseur alternatif. Pour rappel, LLVM est une infrastructure de compilateur open source très répandue.

Elle est conçue de manière modulaire, ce qui facilite l'ajout de compatibilité à de nouveaux langages de programmation et à de nouvelles architectures de processeurs.

Ce compilateur LLVM est inclus dans le dernier SDK CUDA disponible, la v4.1. "L'ouverture de la plate-forme CUDA est une étape importante", a déclaré Sudhakar Yalamanchili, professeur à l'Institut de Technologie de Géorgie et responsable du projet Ocelot, dont le logiciel de cartes a été écrit en CUDA C pour les architectures de processeurs différents, "L'avenir de



l'informatique est hétérogène, et le modèle de programmation CUDA fournit un moyen puissant pour tirer le meilleur parti des performances sur les différents types de processeurs, y compris les GPU d'AMD et les processeurs x86 d'Intel. "

Avec cette ouverture, CUDA

cherche à étendre sa communauté et surtout les langages le supportant. Reste à savoir si cette opération va être bénéfique à cette technologie. Réponse dans quelques mois...

Pour en savoir plus :

<http://developer.nvidia.com/category/zone/cuda-zone>

webOS sera open source



Cela fait des mois que HP change d'avis sur son système embarqué / mobile, webOS. Au départ, il s'agissait de se séparer de la partie PC grand public, d'arrêter les tablettes et sans doute aussi le webOS. Puis au

fil des semaines, la position de HP est devenue de plus en plus floue, avec notamment plusieurs ventes à prix cassé de sa tablette TouchPad, mais restait à trouver une solution à webOS. Finalement, sauf nouveau changement, le système devient open source et HP continuera à contribuer au projet.

Quel est le but de ce revirement ?

- accélérer [?] le développement du système et de la plateforme
- HP continue à investir en argent et en capital humain
- Avoir une gouvernance transparente et

éviter la fragmentation par des forks mal contrôlés

- Devenir un véritable projet open source
Cependant, lorsque l'annonce a été faite officiellement le 9 décembre dernier, de nombreux détails restaient à éclaircir : mise en place d'une direction du projet, le type exact de licence open source, les procédures de contributions (et qui pourra contribuer). D'autre part, la partie multiplateforme s'appuyait fortement sur PhoneGAP dont l'éditeur a été racheté par Adobe, quel avenir pour cet accord ? Si cette annonce rassure (un peu) sur l'avenir immédiat de webOS, sur le long terme, il faut tout de même se poser des questions : quels constructeurs supporteront ce système et pour quels usages ? Les développeurs vont-ils investir du temps et des développements pour un système dont le parc machine se résume à quasiment rien ? HP et le projet ouvert devront rapidement répondre à ces questions pour espérer jouer un rôle sur le marché des tablettes, ou tout autre...

Site webOS : <https://developer.palm.com/>
Annonce officielle : <http://www.hp.com/hpinfo/newsroom/press/2011/111209xa.html?mtxs=rss-corp-news>

Java et C# coûteux, C++ moins cher

L'éditeur CAST Software a mené une vaste enquête sur 745 applications et 365 millions de lignes de code analysées. L'enquête s'est efforcée de recenser des problèmes dus à de mauvaises conceptions d'architectures, de mauvaises pratiques d'écriture du code, et elle introduit le concept de dette technique. Cette dette technique représente le coût requis pour corriger un défaut, qu'il s'agisse d'une faille de sécurité, ou d'un problème de performance, par exemple.

L'enquête se base sur un coût horaire de 75 dollars. La correction d'une ligne de code reviendrait en moyenne à 3,61 dollars. Mais l'écart entre les différents langages est important. L'étude met en lumière une réalité que l'on cache souvent : le coût pour réparer une ligne de code Java monte à 5,42 dollars, contre 1,26 pour une ligne Cobol ! Cependant, de très nombreux éléments influencent le coût d'une correction, notamment le niveau de compétence et d'expérience du développeur.

Le rapport complet :

<http://www.castsoftware.com/resources/resource/email-campaigns/cast-report-on-application-software-health?gad=HPH>

**ActiveReports 6** à partir de € 520

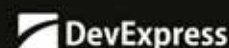
Dernière version du générateur de rapports .NET sans droits le plus vendu.

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

**Janus WinForms Controls Suite V4.0** à partir de € 689

Ajoutez des interfaces de style Outlook à vos applications .NET.

- Vues ruban, grille, calendrier, et barres chronologique/raccourcis
- Nouveau – Style visuel Office 2010 pour tous les contrôles
- Nouveau – Support des profils client Visual Studio 2010 et .NET Framework
- Janus Ribbon ajoute Backstage Menus et la fonctionnalité onglet comme dans Office 2010
- Prend désormais en charge la sélection de cellules multiples

**DXperience DXv2** à partir de € 1 116

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

- Abonnement de 12 mois pour tous les produits et mises à jour DevExpress et accès aux versions bêta en développement actif
- Exploitez vos compétences pour développer des applications tactiles sur toutes les plateformes
- Créez des applications métier avec l'esthétique Windows 8 Metro
- Inclut des fonctions d'aide décisionnelle, de rapport et d'analyse

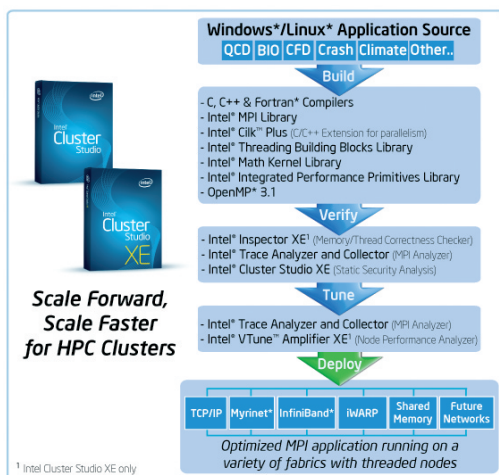
**TX Text Control .NET for Windows Forms/WPF** à partir de € 911

Composants complets de traitement de texte pour Windows Forms et WPF.

- Le traitement de texte professionnel pour vos applications
- Chargez, modifiez et enregistrez au format MS Word (DOCX, DOC, RTF)
- WYSIWYG, tableaux imbriqués, cadres, en-têtes, pieds de pages, images, puces, listes numérotées, zoom, sauts de section, etc.
- Exploitez la liaison de données XAML pour lier les barres d'outils et de ruban

» Intel Software

propose une nouvelle suite de développement et d'optimisation : Intel Cluster Studio XE, qui cible les fortes montées en charge, les applications et systèmes HPC. C'est la première fois que Intel propose une telle suite logicielle spécifique au HPC, centrée autour de MPI, librairie bien connue dans le développement multithread et parallèle. Elle comprend Composer XE (compilateurs et bibliothèques), Trace Analyzer and Collector, MPI Library, MPI Benchmarks, VTune Amplifier XE, Inspector XE. Tarifs : à partir de 1 499 \$ pour la version Cluster Studio, à partir de 2 499 \$ pour la version XE. Disponible sur Linux et Windows. Pour en savoir plus : <http://software.intel.com/en-us/articles/intel-cluster-studio-xe/>

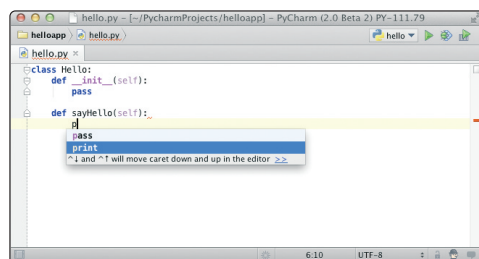


» **Android 4** est maintenant disponible en version 4.0.3, avec indication API Level 15. Cette version inclut de nouvelles fonctions, des changements d'API et des corrections de bug. Les nouveautés API se concentrent sur les contacts, le calendrier, l'écran d'accueil, la pile bluetooth, l'accessibilité, OpenGL ES, les données.

Pour migrer et en savoir plus :

<http://developer.android.com/sdk/android-4.0.3.html>

» **JetBrains** annonce la version 2 de PyCharm. PyCharm est un IDE complet pour Python, Django et Google App Engine. Cette v2 étend le support technique avec l'apparition de Mako, Jinja2, des templates

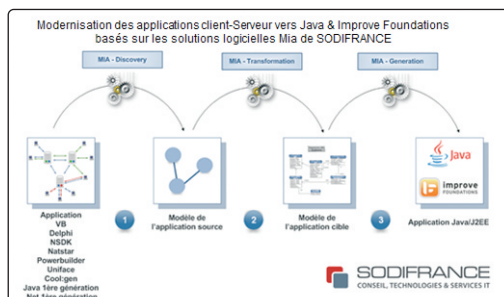


langages. D'autre part, des dialectes JavaScript font aussi leur apparition : CoffeeScript et Cython. Pour les projets open source, l'outil est gratuit. Pour un usage commercial, à partir de 87 €. Site : <http://www.jetbrains.com/pycharm>

» **Opa** est un nouveau langage pour le développement web. Comme les autres langages web, il se veut simple, rapide à maîtriser. Opa S4 apporte une nouvelle syntaxe et le support de NoSQL. Ajax est supporté. Il se veut par défaut distribué pour apporter la souplesse des architectures à la demande, et se veut ouvert sur les offres cloud même : opa-cloud -host localhost -host localhost webchat.exe. A découvrir au moins pour la curiosité : <http://opalang.org/>

» Sodifrance et Open Wide

s'associent pour faire de la modernisation applicative. L'objectif est d'aider les développeurs / entreprises à passer d'une application client/serveur vers un modèle



Java. Pour cela, les deux partenaires s'appuient sur les outils et expertises de chacun : FAST et Improve Foundations. Cette modernisation se fait en 3 grandes étapes (découverte de l'application, la transformation et le modèle moderne que l'on souhaite atteindre) puis la génération.

» ModelioSoft propose une solution

de modélisation et de génération pour C# avec Modelio2. Il s'agit d'un outil de modélisation UML / BPMN et de développement C#. Il se couple à Visual Studio pour les projets .Net. Les développements d'applications .NET bénéficient maintenant d'un support étendu des capacités MDA (architecture guidée par les modèles), d'une modélisation sur toute la portée de l'entreprise. Modelio C# Solution peut se combiner au module de génération documentaire "document publisher", intégrer le support du travail de groupe, être étendu pour gérer les exigences et le dictionnaire, et se combine en outre aux modules de génération tels SQL, XSD, ou WSDL pour fournir une automatisation poussée sur toutes les technologies de développement. A partir de 400 €. Site officiel : www.modeliosoft.com

» Aldebaran Robotics

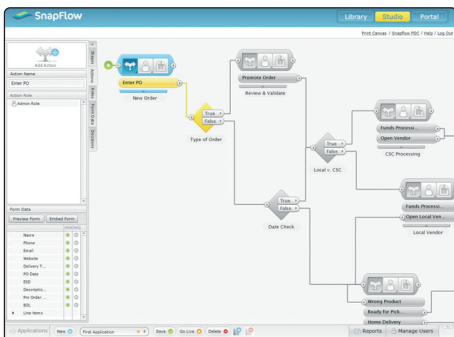
annonce NAO Next Gen.
« L'arrivée de cette nouvelle génération de robots NAO est significative pour notre entreprise. Nous sommes fiers de pouvoir proposer toujours plus à nos clients, quel que soit leur secteur. La maturité du NAO Next Gen va nous permettre de le mettre au service de l'aide aux enfants autistes et aux personnes en



perte d'autonomie. J'ai créé Aldebaran Robotics, en 2005, dans ce but : contribuer au bien-être de l'humanité », déclare Bruno Maisonnier, Président Fondateur d'Aldebaran Robotics, leader mondial de la robotique humanoïde. Les nouveautés sont nombreuses : nouveau processeur Atom, deux caméras HD couplées à un FPGA, nouveau module de reconnaissance vocale (Nuance), nouvelles interactions possibles. Pour en savoir plus : <http://www.aldebaran-robotics.com/>



Silverlight 5 : disponible en version finale, mais après ?



Disponible en préversion finale depuis le mois de septembre, Silverlight 5.0 est désormais en version finale depuis début décembre. Cette version fait le plein de nouveautés : décodage matériel pour H.264, performances en hausse, taille des fichiers améliorée, meilleure animation, meilleure qualité des textes, nouvelles API 3D, XNA, meilleure utilisation de la GPU. Les parties Windows 8 et mobile sont d'ores et déjà

présentes. Cette nouvelle version va de pair avec l'arrivée de nouveaux outils : pack de mise à jour pour Visual Studio, préversion d'Expression Blend. Mais au-delà de Silverlight 5, une question se pose depuis quelques semaines : quel avenir pour Silverlight ? Finalement, à cette question, Microsoft n'a jamais précisément répondu depuis la conférence BUILD de septembre dernier. Il paraît difficile de croire que l'éditeur puisse arrêter cette technologie que l'on retrouve d'une manière ou d'une autre sur Windows 8 et Windows Phone. Mais comme Adobe avec Flash, une redéfinition profonde de la plate-forme va-t-elle être faite ? Aucune nouvelle pour le moment d'une future v5.x ou 6.0.

Site : <http://www.microsoft.com/silverlight>

Qt prépare Windows 8

À défaut de Windows Phone 7.x, Qt se prépare à l'arrivée de Windows 8. En effet, une version adaptée au nouveau Windows sera proposée avec quelques surprises : fonctionnement normal ou quasi en mode desktop, et surtout l'arrivée d'une compatibilité avec l'interface Metro ! Sur ce dernier point, il semblerait que Qt ait oublié de faire des modifications d'objets. Mais cette compatibilité Metro est une nouvelle importante car Qt pourra être utilisé sur tablette et PC. Qt deviendra un « simple » plug-in à WinRT...

Pour le moment, Qt continue à travailler sur la lignée 4.x. Mi-décembre, l'éditeur dévoilait la version 4.8 de la librairie d'interface. Parmi les nouveautés, citons : la pile d'abstraction Qt Platform Abstraction, les threads OpenGL, multithread http et une optimisation de l'accès au système de fichier.

Pour en savoir plus : <http://labs.qt.nokia.com/>



Modelio

Solutions d'entreprise basées sur l'atelier open source leader de modélisation
Modélisation et implémentation des Systèmes et Logiciels

Développeurs logiciels

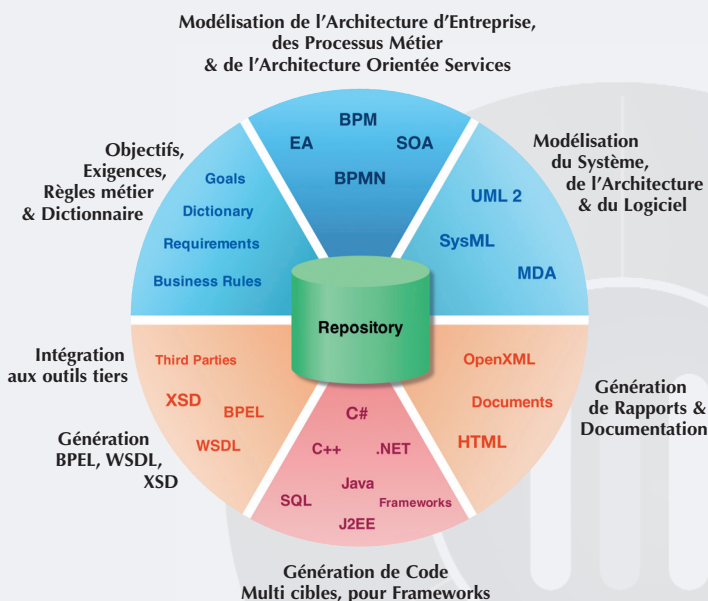
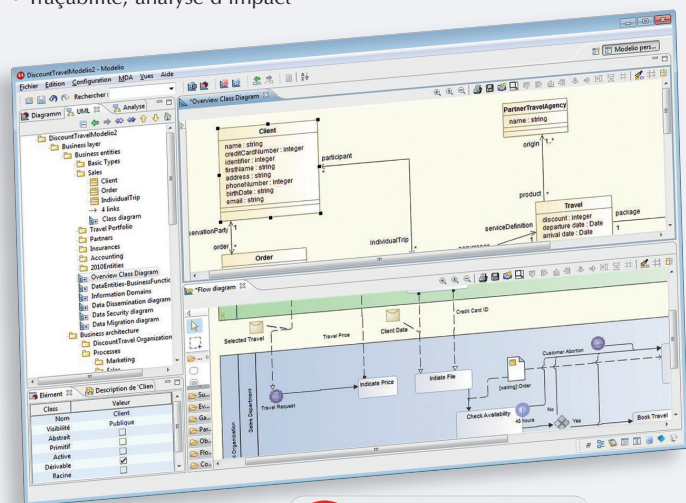
- UML pour Java, C++, C#, SQL, WSDL, XML
- Génération, roundtrip, reverse

Architectes métier

- Modélisation intégrée de UML, BPMN, Architecture d'Entreprise,
- Analyse intégrée des exigences, des objectifs, ...

Architectes système

- Modélisation Systèmes, Logiciels, Hardware, Exigences
- Traçabilité, analyse d'impact



Modelio.org

- L'atelier mature de modélisation UML/BPMN open source
- Le Store Modelio : modules et extensions pour modéliser et générer

Solutions d'entreprise Modelio

- Solutions pour développeurs, architectes métier, architectes systèmes
- Génération documentaire Word, HTML : plans types dédiés
- Travail de groupe distribué, gestion de version et configuration

Explorez votre code Java avec JavaDepend

Préserver un code de qualité, facilement maintenable et évolutif n'est pas une tâche aisée. On ne se rend compte des dégâts que lorsque les bugs s'accumulent, les évolutions deviennent très compliquées et que le téléphone sonne toutes les cinq minutes parce que les clients ne sont pas du tout contents...

Pour éviter d'en arriver au stade où tout le projet entre dans une phase instable, l'équipe de développement doit bien s'outiller. JavaDepend fait partie de ces outils indispensables. Il vous permet d'explorer votre code de diverses façons, apporte bien des fonctionnalités pour l'architecte, le concepteur et le développeur. Examinons ensemble quelques-unes de ses possibilités.

> Le langage CQL et la définition de règles :

CQL permet de requêter le code de la même manière que les requêtes SQL, il est très riche en possibilités et JavaDepend fournit un éditeur de requête CQL simple et intuitif. On peut découvrir toutes les possibilités à partir de cette URL <http://www.javadepend.com/CQL.htm>.

Par exemple, on peut chercher les méthodes les plus utilisées

```
SELECT TOP 10 METHODS ORDER BY MethodCa DESC
```

Ou celles les plus utilisées mais avec une couverture du code insuffisante

```
SELECT TOP 10 METHODS WHERE PercentageCoverage < 80 ORDER BY MethodRank DESC
```

Ou chercher les méthodes les plus compliquées mais mal commentées :

```
SELECT TOP 10 METHODS WHERE CyclomaticComplexity > 20 AND PercentageComment<30
```

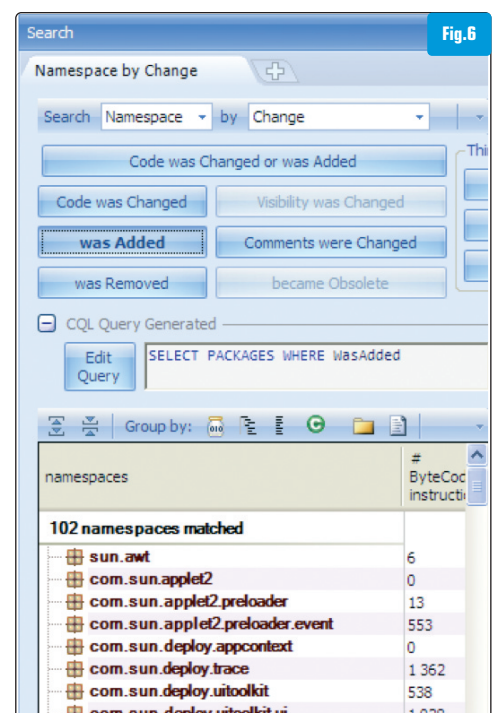
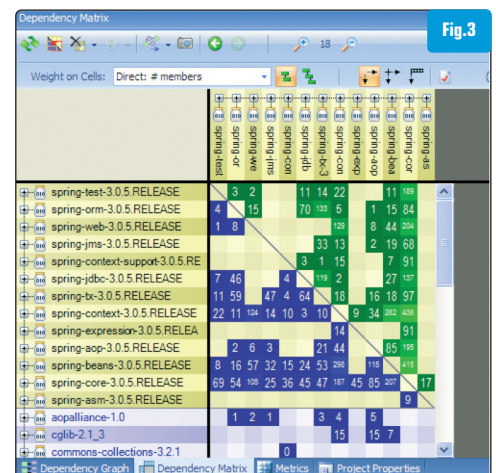
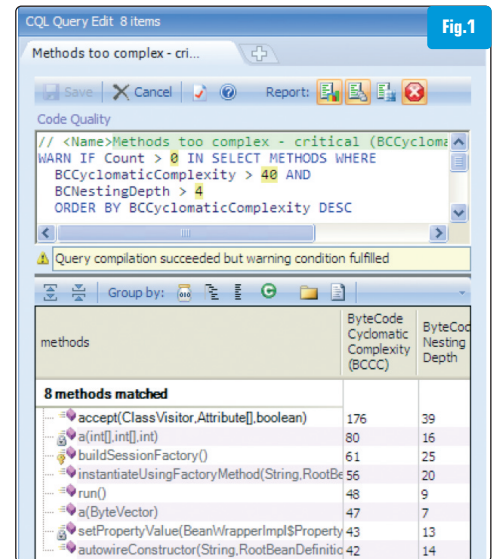
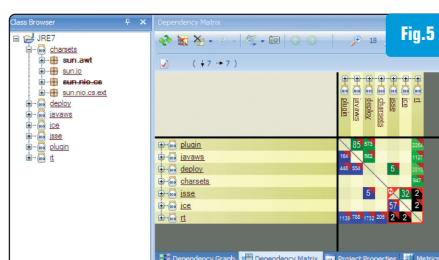
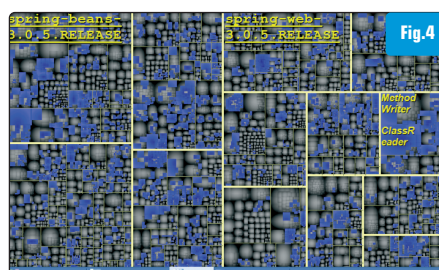
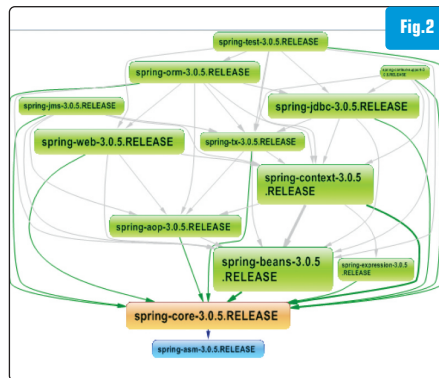
Voici par exemple le résultat d'une requête ciblant le code du framework Spring, le but étant de détecter les méthodes les plus complexes [Fig.1].

Avec le langage CQL on a la possibilité de définir des règles pour contrôler en permanence la qualité du code, plusieurs règles ciblant l'architecture, la conception et l'implémentation sont fournies par défaut, et bien sûr, d'autres règles personnalisées peuvent être ajoutées facilement. De plus, on peut intégrer facilement JavaDepend au niveau de notre process de build, et émettre des avertissements ou erreurs si des règles ne sont pas satisfaites.

> Les vues de dépendances

JavaDepend propose deux vues complémentaires pour les dépendances.

Graphe de dépendances [Fig.2]





Pour cette vue, les rectangles peuvent être proportionnels au nombre de lignes de code, à la complexité, au coupling, au Ranking ainsi qu'à d'autres métriques, de cette manière on peut visuellement avoir une idée plus concrète de notre code.

Vue Matrice [Fig.3]

Cette vue donne plus de détails sur le poids de dépendance entre deux éléments, on peut naviguer à l'intérieur de la matrice et plusieurs fonctionnalités sont fournies pour bien l'exploiter.

Avec cette vue on peut découvrir rapidement quels sont les éléments de code les plus sollicités.

> Métriques et la vue Treemap

JavaDepend propose plus de 82 métriques, qui en les utilisant dans des requêtes CQL nous permettent d'interroger le code plus efficacement.

La vue treemap est un moyen très pratique pour représenter le résultat des requêtes [Fig.4].

Pour cette vue, les classes, les méthodes et les attributs sont représentés sous forme de rectangles, leur taille peut être proportionnelle à la métrique choisie, par défaut ils sont proportionnels au nombre de lignes de code.

Les rectangles en bleu représentent le résultat de la requête, de cette manière on peut voir le code impacté par la requête d'une manière très intuitive, et bien sûr on peut interagir avec cette vue pour avoir plus de détails sur les éléments de code.

> Différence entre versions

Lors de l'évolution du code on a besoin de comparer les versions du même projet, le cas le plus courant est de comparer une version en développement avec une version en production pour détecter les modifications et les breaking changes.

Par exemple pour la comparaison de JRE6 à JRE7, on peut voir le résultat d'une manière visuelle [Fig.5]

Ou utiliser le « SearchPanel » pour avoir plus de détails sur ce qui a changé [Fig.6].

Et de plus, le CQL propose la possibilité de créer des règles concernant la différence entre versions, par exemple on peut chercher les méthodes devenues « deprecated » mais toujours utilisées :

```
SELECT TYPES WHERE BecameObsolete AND TypeCa >0
```

Ou bien les types qui ne sont plus utilisés

```
SELECT TYPES WHERE IsNotUsedAnymore
```

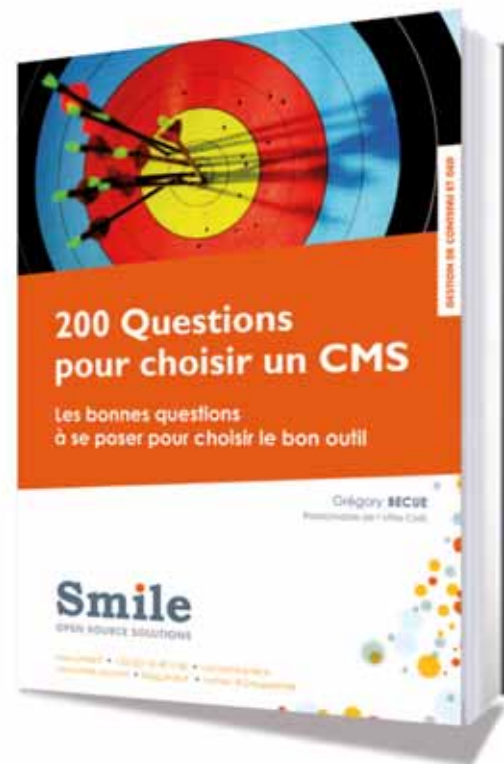
> Conclusion

JavaDepend est un outil complémentaire aux autres outils d'analyses statiques, il apporte une autre vision aux projets Java, et les différentes vues permettent d'explorer d'une manière très intuitive le code source. Le langage CQL apporte une manière élégante de définir des règles pour assurer la qualité de l'architecture, de la conception et de l'implémentation.

Issam Lahlali

Nouvelle version

200 questions pour choisir un CMS



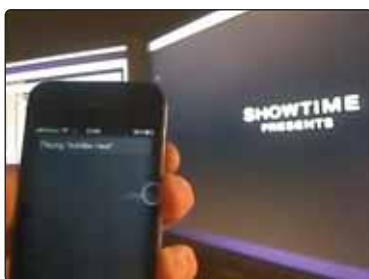
Smile passe au crible
toutes les questions à se poser
pour choisir l'outil le mieux adapté
aux besoins de son projet web

A télécharger gratuitement
sur www.smile.fr

La **réalité augmentée** est un des défis des terminaux mobiles, du web et des applications. Le Français Total Immersion et Texas Instrument viennent de dévoiler une nouvelle plate-forme : D'Fusion pour la plate-forme OMAP du constructeur américain. « D'Fusion se positionne comme la plateforme de réalité augmentée la plus aboutie depuis de nombreuses années, et ce sur l'ensemble des plateformes du marché dont les plateformes mobiles. La robustesse de OMAP et l'étendue de la communauté de développeurs qui l'utilisent au quotidien nous encouragent à nous concentrer sur l'optimisation de la RA avec TI » précise Bruno Uzzan, Co-Fondateur et CEO chez Total Immersion. « Nous sommes très heureux de travailler avec TI pour repousser les limites de la RA sur les Smartphones, les tablettes et autres dispositifs mobiles. La plateforme D'Fusion Computer Vision optimisée pour les processeurs mobiles OMAP proposera aux développeurs de RA les fonctions les plus avancées dans un environnement facile d'utilisation. »



Toyota a dévoilé un « concept car » pour le moins original et étonnant : la Fun-Vii. Ni avant ni arrière, on ne sait comment elle se présente. Toyota veut une voiture totalement interactive, connectée aux réseaux. Tout ou presque peut servir d'écran, d'interaction. Il est possible de personnaliser la carrosserie en ajoutant une photo. Elle dispose de 3 places et est électrique. Un pur délice !



L'assistant vocal d'Apple, Siri, disponible sur l'iPhone 4S, passionne les développeurs et autres bidouilleurs. Un exemple parmi d'autres, le projet Siri Proxy, et sa passerelle avec le media center open source Plex, permet de transformer

son 4S en télécommande vocale pour sa télévision ! Apple a annoncé l'ouverture de quelques API Siri dans iOS 5.1. Cette technologie pourrait rapidement devenir une vedette !

Siri Proxy : <https://github.com/plamoni/SiriProxy>

Et aussi la télécommande en vidéo :

http://www.youtube.com/watch?v=eChSxAxcxUE&feature=player_embedded#



La poussette n'est pas très tendance ni "nouvelles technologies" ? 4Moms lance **Origami**, une poussette 2.0 pourrait-on dire : écran LCD, lumières incorporées et surtout une capacité à se plier et déplier d'un bouton, fini le casse-tête de la faire à main ! A partir de janvier, prix de vente : environ 850 \$.

Site : <http://www.4moms.com/origami>

Roboden est un câble USB élastique que l'on peut étendre jusqu'à 1,5 fois. Il se compose de deux éléments : une gaine élastique et un câble USB en spirale pour se tendre, se détendre. Cela pourrait éviter des câbles trop longs et offrir de belles perspectives dans la robotique ou les vêtements connectés.

Runware lance un nouvel ensemble pour les amateurs de jogging : runalyzer. L'outil se présente sous la forme d'une « clé » que l'on connecte à son terminal iOS. Elle fonctionne avec les principales ceintures cardio-analogiques du marché. Le Runalyzer est adapté pour les sportifs débutants ou confirmés qui souhaitent mesurer les effets de leur entraînement et suivre leurs progrès. Il est idéal pour les coureurs occasionnels et les personnes qui veulent reprendre une activité physique. Le Runalyzer est un petit récepteur électronique de 4 grammes qui se connecte à l'iPhone ou à l'iPod touch et qui permet au sportif de s'affranchir de la montre tout en étant compatible avec la ceinture cardio qu'il possède déjà. L'ouverture de l'API Runalyzer permet à tous les développeurs iOS d'intégrer facilement cette composante physiologique dans leurs applications. Pour compléter sa gamme, Runware propose en option une ceinture cardio textile et un brassard.



➔ Vous avez bricolé, créé un matériel, fait une utilisation originale ? La rubrique « matériel » vous est ouverte ! Contactez-nous sans plus tarder : redaction@programmez.com

IBM mise sur les puces de nouvelle génération pour améliorer les performances des téléphones et des ordinateurs. Le fondeur n'y va pas par quatre chemins et annonce une rapidité x15 et une taille 90 % plus petite ! Le but : réduire la consommation, l'espace utilisé, tout en boostant les performances. Micron sera le premier à utiliser ce procédé IBM pour offrir un nouveau type de mémoire avancée – l'Hybrid Memory Cube 3D (HMC) destiné aux smartphones et PC. L'HMC offre une bande passante et une efficacité largement supérieures aux capacités des dispositifs actuels. Les prototypes HMC, par exemple, offrent un débit de 128 gigaoctets par seconde (Go/s). Pour comparaison, les dispositifs actuels de pointe plafonnent à un débit de 12,8 Go/s. Pour en savoir plus, le site du consortium HMC : <http://www.hybridmemorycube.org/>

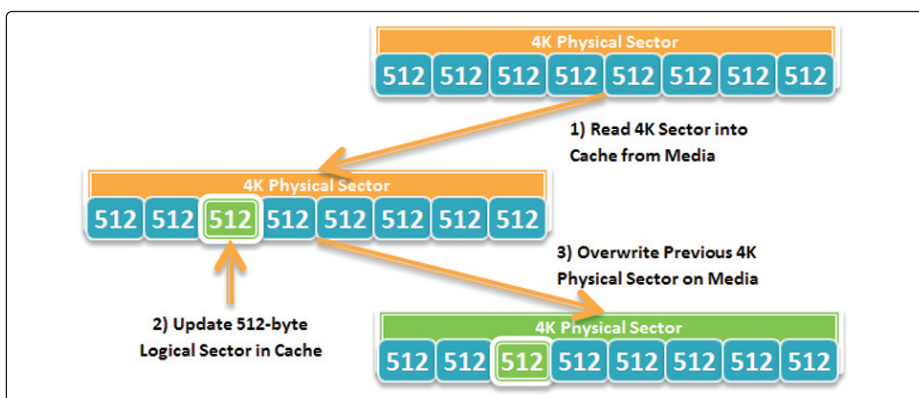
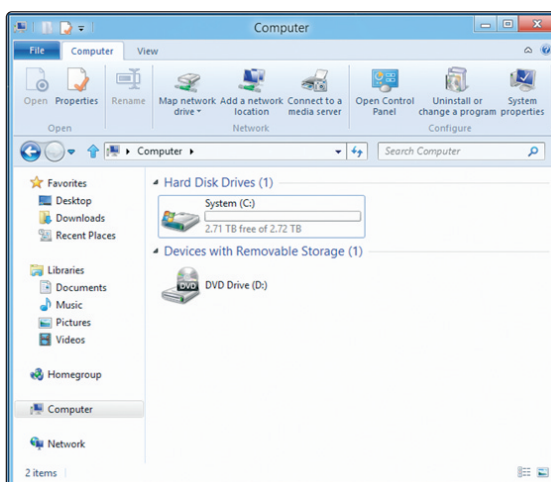
Si **ChromeOS** a bien du mal à exister sur le marché, notamment par la faible offre matérielle adaptée, il existe un projet indépendant pour rendre disponible la version open source, ChromiumOS, sur d'autres modèles de netbooks. Le projet Lime « permet » cela même si les premières versions étaient très limitées sur la partie réseau et le support matériel. Le développeur annonce le support de plusieurs cartes réseaux par défaut (Realtek et Broadcom) et de nouveaux GPU. Si vous voulez tenter l'aventure, c'est ici : <http://chromeos.hexxeh.net/lime.php>



Par contre, un autre projet a sans doute plus d'avenir que le précédent : **Android to x86**. Il s'agit d'un portage d'Android sur les processeurs x86 pouvant faire fonctionner le système mobile Android sur des PC. Le travail avance mais le projet est très loin d'être finalisé. Si pour le moment l'intérêt semble limité, dans le futur, la possibilité de disposer d'Android sur PC ne serait pas aussi absurde que cela, et pourquoi pas, inciter Google à faire une édition officielle... Pour tester : <http://www.android-x86.org/>

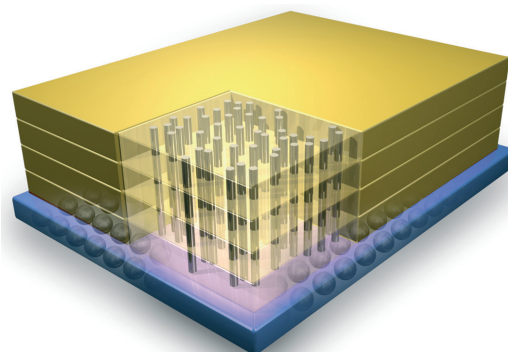


Microsoft se prépare à un petit bouleversement : Windows 8 supportera nativement les (très) gros disques durs. C'est-à-dire les disques de 3 To et plus. Cette évolution est importante avec l'accroissement de la capacité des disques, en 2015, IDC prédit des stockages de 8 To, plus du double qu'actuellement ! Mais cette prise en charge nécessite des modifications techniques et architecturales non négligeables au cœur même de Windows, pour pouvoir adresser l'ensemble de la capacité et assurer le support des différents modèles du marché (ayant des secteurs de large taille). Cela implique, selon l'équipe technique de Windows 8 d'adopter une nouvelle méthode d'adressage disque, de disposer d'un schéma de partitionnement adapté et enfin d'un firmware capable de supporter ces capacités (que se soit un bios ou un UEFI). Le post officiel est particulièrement intéressant à lire : <http://blogs.msdn.com/b/b8/archive/2011/11/29/enabling-large-disks-and-large-sectors-in-windows-8.aspx>



Un laboratoire de l'école polytechnique fédérale de Lausanne vient de dévoiler la première puce électronique réalisée non pas en silicium mais dans un nouveau minéral, la **molybdénite**. Ce minéral permet de dépasser les limites physiques du silicium. Par exemple, sur l'épaisseur, le nouveau minéral peut descendre à 0,65 nm, contre 2 nanomètres pour le silicium ! « Nous avons créé un premier prototype mettant deux à six transistors en série et démontré que des opérations de logique binaire basiques étaient possibles, ce qui prouve que nous pourrions réaliser des circuits plus importants », explique Andras Kis, responsable du LANES, qui a récemment publié deux articles sur le sujet dans le journal «ACS Nano». Comme le précise l'annonce officielle, au début 2011, ce laboratoire avait dévoilé le haut potentiel

du sulfure de molybdène (MoS₂), un minéral que l'on trouve en grande quantité à l'état naturel. Sa structure et ses propriétés de semi-conducteur en font un matériau idéal pour la réalisation de transistors. Il concurrence ainsi directement le silicium, composant le plus utilisé dans le domaine de l'électronique, et aussi en plusieurs points le graphène.



Les Rapports DevExpress



PRESENTATION CONTROLS | REPORTING CONTROLS
BUSINESS APP FRAMEWORKS | IDE PRODUCTIVITY TOOLS



XtraReports Suite

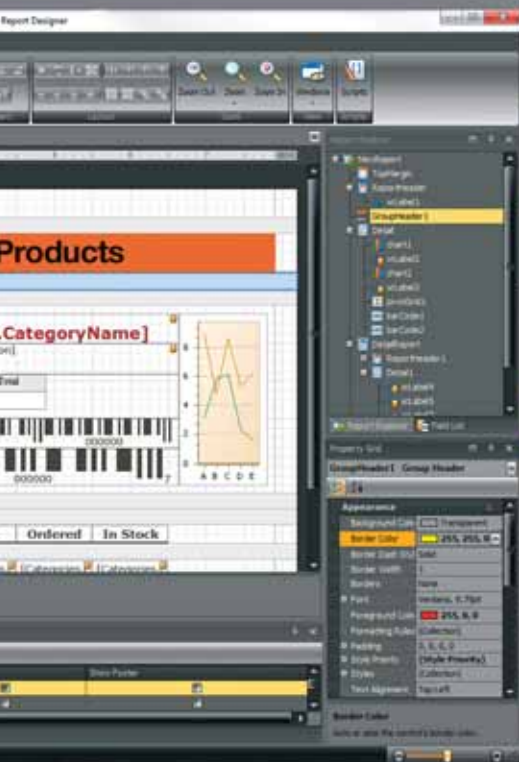
Cross-Platform .NET Support

DevExpress crée des contrôles de présentation aux fonctionnalités complètes, des systèmes de rapports, des outils de productivité pour IDE et des frameworks d'applications Business pour Visual Studio .Net. Nos technologies vous aident à construire ce qu'il ya de meilleur, à avoir une vision plus claire des logiciels complexes, à améliorer votre productivité et à créer, dans le temps le plus court, des applications étonnantes pour Windows et pour le Web.

XtraReports Suite, de DevExpress, est une plateforme de Reporting de nouvelle génération pour Visual Studio. Elle vous permet de créer très rapidement des rapports professionnels, au pixel près, ciblant toutes les plateformes majeures Windows, incluant **WinForms**, **ASP.Net**, **Silverlight**, **WPF**. Au-delà de son ensemble de fonctionnalités de niveau professionnel, XtraReports Suite est livré avec un Designer de Rapports ergonomique pour l'utilisateur final, vous permettant de répondre avec la plus grande flexibilité aux exigences de Reporting.

Téléchargez aujourd'hui votre version gratuite d'évaluation et faites l'expérience de la Différence DevExpress.

www.DevExpress.com/Reporting



DevExpress[™]

WWW.DEVEXPRESS.COM

Les failles du Web : petite analyse d'un site web

Extrait de Sécurité informatique - Ethical Hacking

“ Un site Web est constitué d'une multitude de pages qui sont obligatoirement organisées suivant une arborescence. C'est le plan du site. Chaque page est généralement accessible par un lien hypertexte présent dans un menu ou sur une page. Chaque lien pointe vers une URL ou URI précisant l'endroit où se trouve la ressource. De plus, des médias sont souvent présents dans les pages et là encore il existe une URI pour chacun d'eux. Nous pouvons déjà à ce stade distinguer deux types de ressources, celles qui sont dans le même domaine que le site consulté et celles qui sont dans un autre domaine. Si nous voulons faire une cartographie du site, c'est-à-dire lister tous les éléments qui le composent, nous devons nous limiter au domaine où se trouve le site, car dans le cas contraire nous allons parcourir tout l'Internet, étant donné la multitude de liens qui existent entre les pages Web.

Il est même parfois difficile de lister l'ensemble des pages d'un même domaine. En effet, depuis l'avènement de sites dynamiques, la construction des pages se fait en allant chercher de multiples informations sur de nombreuses sources. Ce phénomène multiplie le nombre de pages disponibles et celles-ci ne sont plus qu'une représentation de l'information. Par exemple, dans un site présentant des relevés météorologiques, les pages sont construites en allant chercher l'information dans la base de données des relevés. Si ceux-ci existent depuis 10 ans et que nous pouvons en demander une représentation par jour, mois ou année, le nombre de pages pouvant être généré devient impressionnant. Ce n'est donc pas la bonne méthode que de tenter d'analyser toutes les pages. En fait, il vaut mieux essayer d'analyser un comportement, il faut trouver le maximum d'informations que le site veut bien nous donner dans un fonctionnement normal. Nous pouvons alors plutôt parler de prise d'empreinte que de cartographie.

Voici un petit listing des questions que nous pouvons nous poser afin de recueillir le maximum d'informations :

- Le site est-il statique ou dynamique et dans ce cas, dans quel langage est-il écrit ?
- Quelles sont les variables utilisées pour transmettre les requêtes ?
- Y a-t-il des formulaires et quels champs utilisent-ils ?
- Recevons-nous des cookies, quelles données contiennent-ils ?
- Les pages contiennent-elles des médias ?
- Le site fait-il appel à des bases de données ?
- Pouvons-nous accéder à des dossiers, contenant des images par exemple ?
- Le site fait-il appel à du JavaScript, de l'AJAX ?
- Quel serveur est utilisé, quelle est sa version ?

Cette liste, qui n'est pas exhaustive, permet déjà de rassembler un grand nombre d'informations. Reprenons chaque point et voyons comment nous pouvons tenter d'obtenir une réponse.

> Le site est-il statique ou dynamique ?

La première chose à regarder est l'extension du fichier appelé dans l'URL lors de la navigation. Si celle-ci est du type .php ou .asp ou

encore .jsp, c'est que le site est écrit dans un de ces langages, sauf exception rarissime si un petit malin a voulu nous tromper en modifiant les réglages de son serveur. Si l'extension est du type .html ou .htm c'est probablement une page statique, mais attention ici rien n'est moins sûr. En effet les serveurs permettent de faire de l'URL Rewriting (réécriture d'adresse). Cette technique permet un meilleur référencement des sites et masque le passage des variables ainsi que le type de langage utilisé.

Par exemple si nous appelons une page php :

```
pa_ges.php?id=5&menu=2&article=7
celle-ci peut être réécrite comme suit :
pa_ges_5_2_7.html
```

Nous constatons quand même que la page porte un drôle de nom, ce qui peut nous mettre sur la piste.

Si l'URL ne nous donne aucune information, nous pouvons nous lancer dans l'analyse du code source de la page. Dans Firefox la combinaison des touches [Ctrl]+[U] affiche ce code.

Analysons par exemple le code source de la page d'accueil du site d'ACISSI (<http://www.acissi.net>) dont l'URL ne nous donne pas beaucoup d'informations :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"> <html xmlns="http://www.w3.org/1999/xhtml" lang="fr" xml:lang="fr">
...
...
...
<!--//--><![CDATA[//><!--
jQuery.extend(Drupal.settings,{« base Path » : « / », «htmlMenu » :
{ [ "double click", "clone" ],"googleanalytics": ...
```

Nous remarquons que le code de la page est du XHTML 1.0 Strict et voyons apparaître un peu plus bas le mot Drupal. Ce mot nous parle, car Drupal est un CMS (système de gestion de contenu) écrit en PHP. Comme en naviguant sur le site d'ACISSI nous ne voyons pas de pages avec l'extension php, nous pouvons en déduire que le site utilise de la réécriture d'adresse.

Dans le cas où nous n'arrivons pas à découvrir le langage utilisé pour écrire le site que nous étudions, nous aurons peut-être cette information en analysant les autres points.

```
"base Path":
```



> Quelles sont les variables utilisées ?

Dans un site Web dynamique il n'existe pas autant de scripts que de pages, sinon cela n'aurait aucun sens. Donc un script donné va être capable de construire une multitude de pages en fonction des informations qu'il va recevoir. Celles-ci peuvent être transmises dans l'URL en utilisant la méthode GET. Une analyse de l'URL permet de lister les variables utilisées. Par exemple :

```
forum.php?id=234&user=codej
```

Nous appelons ici le script « forum.php » en passant des variables « id » et « user ». Si une réécriture d'adresse a été mise en place sur le serveur, il devient alors difficile de déterminer le nom des variables.

> Y a-t-il des formulaires et quels champs utilisent-ils ?

Une autre façon de transmettre des données au script est l'utilisation des formulaires. C'est alors la méthode POST qui est généralement utilisée, mais ce n'est pas une obligation. Pour connaître ces champs, nous pouvons regarder le code source ou espionner la transmission des données au serveur comme nous l'avons déjà fait avec « Wireshark ». Nous verrons un peu plus loin qu'il existe des outils bien plus pratiques, mais il est important d'avoir une bonne vision des choses. C'est pour cela que nous vous proposons régulièrement de regarder le code source de la page pour bien comprendre les mécanismes qui sont mis en jeu. Regardons par exemple le code du formulaire suivant :

```
<form action="identif2.php" method="POST">
<input type="hidden" name="type"
value="7d4d77af8208a8b7fc9a2246d97db964">
Votre message : <input type="text" name="message"><br>
<input type="submit" value="Valider">
</form>
```

Nous voyons dans ce formulaire trois champs :

- Un champ de type hidden de nom type qui est donc caché et qui n'apparaîtra pas aux yeux de l'internaute.
- Un champ de type text de nom message qui correspond à un champ visible pour l'utilisateur dans lequel il peut saisir du texte.
- Un champ de type submit qui correspond au bouton de validation du formulaire.

Nous voyons aussi qu'à la soumission du formulaire, celui-ci va transmettre ses données au script identif2.php par la méthode POST.

> Le serveur envoie-t-il des cookies ?

Les cookies sont de petits fichiers contenant du texte que le serveur peut envoyer au client afin de stocker des informations lui permettant de mémoriser une situation. Un des types de cookies les plus couramment utilisés est le cookie de session. Ceux-ci interviennent dans le mécanisme d'authentification du client pour éviter à l'internaute de saisir de nouveau son identifiant/mot de passe à toutes les pages. Il permet en principe au serveur d'identifier le client de façon univoque. Mais ce n'est pas la seule utilisation des cookies. Ceux-ci peuvent aussi permettre de mémoriser vos habitudes, une configuration, etc.

Dans Firefox, une des façons de regarder si le site envoie des cookies est d'aller dans le menu Outil puis de cliquer sur Informations sur la page.

> Le site contient-il des médias ?

Nous pouvons, comme pour les cookies, lister les médias présents sur la page dans l'onglet Médias de la fenêtre Information sur la page de Firefox. Nous avons même la possibilité dans cet onglet de sauvegarder tous les médias, mais surtout nous voyons le chemin de chacun d'eux, ce qui peut être très intéressant.

> Le site fait-il appel à des bases de données ?

Ce point est beaucoup plus délicat à traiter. Nous ne pouvons pas voir en première approche si le site utilise des bases de données. Suivant la nature du site nous pouvons avoir de fortes présomptions mais pour en obtenir la preuve il faudra faire appel à des notions et outils plus avancés que ceux que nous avons vus actuellement. Nous pouvons néanmoins dire que si le site est dynamique, il y a de grandes chances pour qu'il utilise une base de données.

> Pouvons-nous accéder à certains dossiers ?

Généralement nous ne pouvons pas lister le contenu des dossiers contenant les images et autres fichiers, sauf si cette fonctionnalité est volontairement autorisée. Mais il arrive que les serveurs soient mal configurés et autorisent la visualisation du contenu de dossiers sensibles. Nous pouvons avoir une idée des chemins à tester en regardant la source des médias ou d'autres fichiers que le site utilise. Par exemple si nous regardons les médias de la page d'accueil d'ACISSI nous avons une image dont le chemin est : http://www.acissi.net/sites/default/themes/acissi_v2/images/fond1.jpg. Mais si nous tentons d'accéder au dossier www.acissi.net/sites/default/themes/acissi_v2/images nous avons en retour Forbidden qui correspond à l'erreur 403, soit "refus de traitement de la requête". Par contre, si nous essayons de lister le dossier d'un miroir "Debian", comme par exemple : <http://cdimage.debian.org/debian-cd/5.0.1/386/iso-cd/>, cette fois cela fonctionne, ce qui est normal puisque autorisé volontairement.

Toute l'astuce ici consiste donc à lister un dossier pour lequel le concepteur du site a oublié de régler une interdiction.

> Le site fait-il appel à du JavaScript ?

Pour savoir si le site fait appel à du JavaScript, il suffit de regarder le code source. Deux cas sont possibles, soit le code est directement écrit dans la page, soit il est dans un fichier séparé et est chargé par la page. Dans tous les cas, comme notre navigateur doit pouvoir accéder à ce code nous pouvons le récupérer pour l'analyser. Par exemple au début du code source de la page d'accueil d'ACISSI nous trouvons :

```
<script type="text/javascript" src="/misc/jquery.js?U"></script>
<script type="text/javascript" src="/misc/drupal.js?U"></script>
<script type="text/javascript"
src="/sites/default/files/languages/fr_816ac8cfa9d87c1a57c49584baa
730fe.js?U"></script>
<script type="text/javascript" src="/sites/all/modules/dhtml_menu/
dhtml_menu.js?U"></script>
<script type="text/javascript"
src="/sites/all/modules/lightbox2/js/lightbox.js?U"></script>
<script type="text/javascript">
<!--//--><![CDATA[//><!--
```

```
jQuery.extend(Drupal.settings, { "basePath": "/", "dhtmlMenu":
[ "doubleclick", "clone" ], "googleanalytics": { "trackOutgoing":
1, "trackMailto": 1, "trackDownload": 1,
"trackDownloadExtensions":
"7z|aac|avi|csv|doc|exe|flv|gif|gz|jpe?
g|js|mp(3|4|e?g)|mov|pdf|phps|png|ppt|rar|sit|tar|torrent|txt|wma|
wmv|xls|xml|zip" }, "lightbox2": { "rtl": "0", "file_path": "/
(\\w\\w)/sites/default/files", "default_image":
"/sites/all/modules/lightbox2/images/brokenimage.jpg",
"border_size": 10, "font_color": "000", "box_color": "fff",
"top_position": "", "overlay_opacity": 0.8, "overlay_color":
"000", "disable_close_click": true, "resize_sequence": 0,
"resize_speed": 400, "fade_in_speed": 400, "slide_down_speed":
600, "use_alt_layout": false, "disable_resize": false,
"disable_zoom": false, "force_show_nav": false, "loop_items":
false, "node_link_text": "View Image Details", "node_link_target":
false, "image_count": "Image !current of !total", "video_count":
"Video !current of !total", "page_count": "Page !current of !
total", "lite_press_x_close": "press \\x3c a href=\\\"#\\\"
onclick=\\\"hideLightbox(); return FALSE;\\\"\\x3e\\x3c kbd\\
x3e\\x3c a\\x3e to close", "download_link_text": "", "enable_login":
false, "enable_contact": false, "keys_close": "c x 27",
"keys_previous": "p 37", "keys_next": "n 39", "keys_zoom": "z",
"keys_play_pause": "32", "display_image_size": "",
"image_node_sizes": "()", "trigger_lightbox_classes": "",
"trigger_lightbox_group_classes": "", "trigger_slideshow_classes":
"", "trigger_lightframe_classes": "", "custom_class_handler": 0,
"custom_trigger_classes": "", "disable_for_gallery_lists": true,
"disable_for_acidfree_gallery_lists": true,
"enable_acidfree_videos": true, "slideshow_interval": 5000,
"slideshow_automatic_start": true, "slideshow_automatic_exit":
true, "show_play_pause": true, "pause_on_next_click": false,
"pause_on_previous_click": true, "loop_slides": false,
"iframe_width": 600, "iframe_height": 400, "iframe_border": 1,
"enable_video": false } } );
//-->]]>

</script>
<script type="text/javascript"> </script>
<!--[if lte IE 6]>
<script type="text/javascript">
  $(document).ready(function() {
    $(document).pngFix();
  });
</script>
<![endif]-->

<!--<link rel="alternate" type="application/rss+xml"
title="ACISSI" href="/rss.xml" />-->
```

Nous remarquons une combinaison de scripts enregistrés dans des fichiers séparés, comme lightbox.js, et de JavaScript directement écrits dans la page. Nous pouvons récupérer le code des fichiers JavaScript en les appelant dans l'URL du navigateur, par exemple : <http://www.acissi.net/sites/all/modules/lightbox2/js/lightbox.js>

Sous Linux nous pouvons aussi récupérer le fichier grâce à la commande wget :

```
wget www.acissi.net/sites/all/modules/lightbox2/js/lightbox.js
```

Dans le cas du site d'ACISSI nous aurions aussi très bien pu récupérer l'ensemble des scripts en téléchargeant le code source de Drupal qui est un CMS libre. L'ensemble des scripts est donc fourni lorsque nous téléchargeons Drupal.

> Quel serveur est utilisé et quelle est sa version ?

Avoir une connaissance précise du type de serveur et de son numéro de version est très important car il existe des failles de sécurité référencées pour chacun d'eux. Il n'est pas rare de trouver cette information dans l'en-tête de la réponse du serveur. Une autre façon d'obtenir ces informations est de provoquer le renvoi de la page d'erreur, par exemple en appelant une page qui n'existe pas sur le serveur. Celle-ci peut être appelée dans l'URL du navigateur ou à l'aide d'un logiciel pour avoir le détail de l'erreur renvoyée.

Nous pouvons utiliser par exemple Netcat.

> À l'aide

Nous constatons que tous ces tests sont longs et fastidieux mais c'est une bonne école. Heureusement pour nous, il existe des outils qui peuvent nous faciliter la vie. Un add-on bien intéressant de Firefox est Web Developer que nous trouvons à l'adresse <https://addons.mozilla.org/fr/firefox/addon/60>. Une fois cet add-on installé et Firefox redémarré nous voyons apparaître une nouvelle barre d'outils qui va nous permettre d'effectuer toutes les opérations que nous avons vues précédemment et bien plus encore

- pour les formulaires : Forms => View Form Information ;
- pour les cookies : Cookie => View Cookie Information ;
- pour le JavaScript : Information => View JavaScript ;
- pour l'en-tête : Information => View Response Headers ;
- etc.

Extrait de Sécurité informatique - **Ethical Hacking**
Apprendre l'attaque pour mieux se défendre [2^e édition],
de M. Hennecart et Ebel Franck, éditions ENI.

L'information permanente

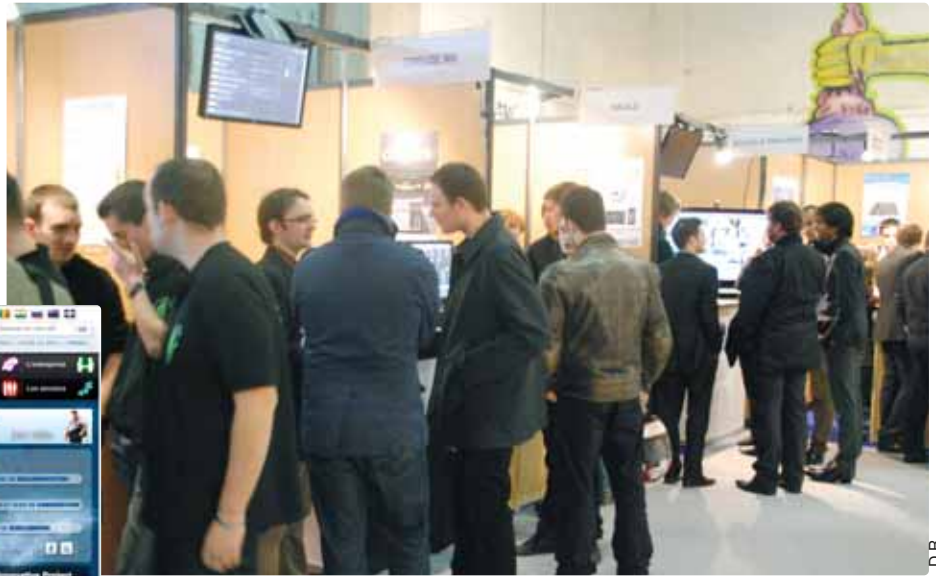
The screenshot shows the Programmez.com website with a navigation bar at the top. The main content area features several articles, including one about Windows 8 and another about the security of the generation. There are also advertisements for Arkeia Network Backup and FLEX SERVER. The website has a clean, professional layout with a mix of text and images.

- L'actu de Programmez.com : le fil d'info quotidien
- La newsletter hebdo : la synthèse des informations indispensables. Abonnez-vous, c'est gratuit !

www.programmez.com

La folie des projets d'études de l'Epitech !

Quand les étudiants innovent...



Une nouvelle génération de développeurs et d'entrepreneurs expose leur savoir-faire



Une des spécificités de l'Epitech est que dès la 3e année du cursus d'études, les étudiants doivent rejoindre un groupe projet, Epitech Innovative Project (EIP). Celui-ci va suivre l'étudiant durant deux ans jusqu'à la fin de son cycle. Puis, parfois, va même déboucher sur une création d'entreprise...

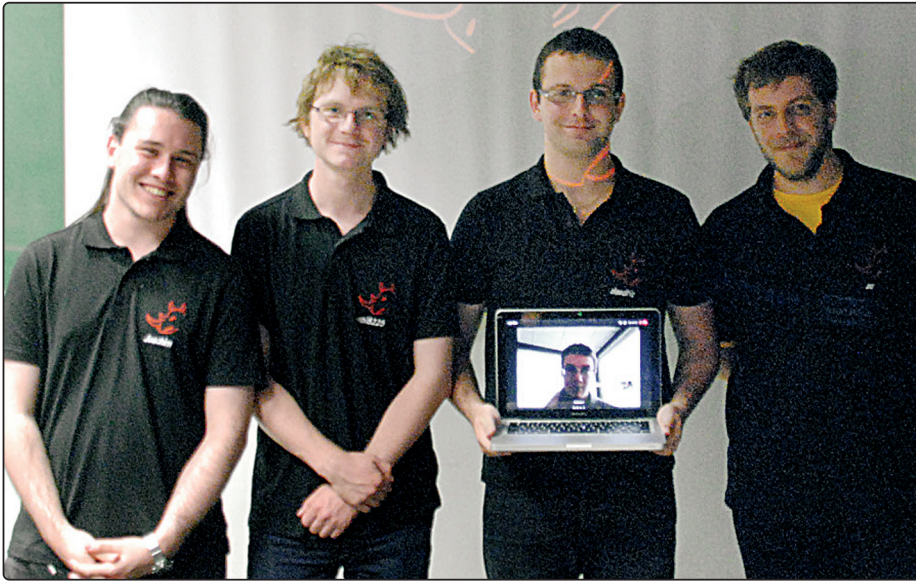
Ces EIP sont pour les étudiants une mise à l'épreuve réelle : gestion de projet, choix technique, gestion des ressources humaines. Et tout ne va pas de soi ! Car deux ans c'est long et court à la fois. Il ne faut pas perdre de temps.

Le niveau de chaque projet est parfois inégal mais il y a toujours quelque chose à en tirer ou à apprendre.

Chaque année, sur le campus, un grand événement expose l'ensemble des 52 EIP en cours rassemblant pas moins de 330 étudiants ! Parmi tous ces projets, nous en avons sélectionné 6. Ce choix fut difficile car de nombreux EIP mériteraient d'être présentés dans ce dossier spécial. Bonne découverte !

Reportage réalisé par François Tonic

Rathaxes, le générateur



Nous avons rencontré l'équipe de Rathaxes en 2008 et le projet nous avait beaucoup plu. Trois ans plus tard, les étudiants ont changé mais le projet demeure et a même beaucoup évolué. L'idée de base est ambitieuse : « *Rathaxes est un langage dédié qui permet de faciliter le développement de pilote de périphériques; la partie logiciel qui fait fonctionner les composants électroniques de votre ordinateur* ». Rathaxes est un DSL (Domain Specific Language) permettant de décrire le pilote matériel qui sera ensuite généré. Surtout, le prototype 2011 est aujourd'hui fonctionnel. Il est écrit en C et génère le code pilote pour Linux, Windows 7 et OpenBSD. Il supporte une plus grande variété de périphériques (ex. : carte réseau, souris USB).

La nouvelle équipe veut ouvrir Rathaxes à la communauté afin que l'outil puisse évoluer et attirer les contributions. Si la plupart des projets de fin d'études débouche sur quelque chose (mise à disposition en open source, projet commercial, reprise par un nouveau groupe), Rathaxes est un peu particulier car il y a une importante partie de recherche. L'absence de réelle offre concurrente rend l'exercice encore plus difficile pour les étudiants, même si c'est en même temps motivant.

La première mission de l'équipe fut de concevoir un modèle de génération le plus cohérent possible. Le premier prototype s'était essentiellement concentré sur les notions de langages et de générations. Un énorme travail de réflexion, de recherche a été réalisé

sur la partie description du matériel et système. Durant deux ans, les étudiants ont dû résoudre le problème de l'implémentation de Rathaxes : n'étant pas un compilateur « standard », il fallait définir le langage avant d'espérer arriver à un modèle de description. Parfois, les développeurs ont dû faire marche arrière comme sur le paradigme aspect où des fonctions furent tout simplement zappées, trop compliqué. Le développement système est un monde à part...

> Architecture du projet

Basiquement, Rathaxes se compose de :

- un langage DSL
- une black-librairie : assure l'interface entre le langage et le compilateur

- compilateur : transforme le fichier .rtx en fichier .c spécifique à la plate-forme cible. L'architecture repose essentiellement sur les différentes passes de compilation et donc sur ses modèles :

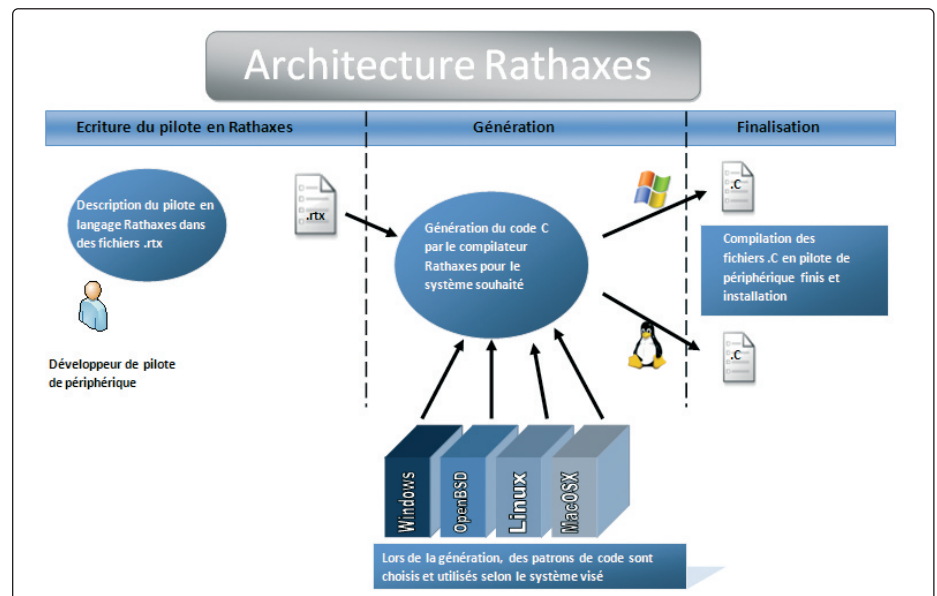
- parsing : contient les règles d'analyse syntaxique pour les différentes parties du DSL
- cache : « base de donnée » des modèles de codes pré-compilés,
- module de typage : assure les vérifications de typage pour confirmer la cohérence du fichier analysé
- module de génération (template) : génère le code de résolution du template
- module de génération : utilise les informations du front-end pour lancer la génération du pilote basée sur une résolution récursive des modèles de codes utilisés.

> Quel avenir ?

Tout le monde souhaite qu'il puisse continuer, le travail restant à accomplir est énorme. Malheureusement, en France, le développeur système n'est pas reconnu, ou si peu. Nos étudiants devront donc se tourner vers d'autres secteurs comme la finance. Espérons qu'une belle communauté puisse un jour reprendre et développer le projet.

Pour en savoir plus :

<http://code.google.com/p/rathaxes/>

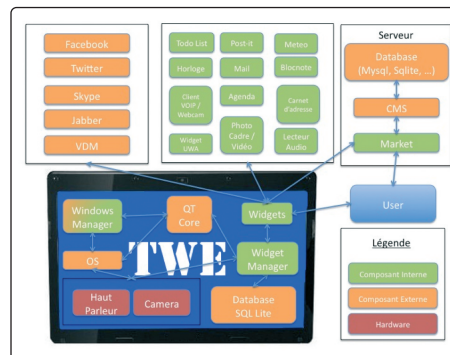


Tactile Widget Environnement

Vive l'embarqué !

Le tactile est partout, envahit tout. Le projet Tactile Widget Environnement (TWE) a pour but de fournir un terminal tactile sous environnement Linux avec une interface adaptée. Les marchés visés sont très vastes : cuisines, communication, aide à la personne, domotique.

Le projet utilise une interface graphique spécifiquement développée avec une API pour



pouvoir développer rapidement. Le but ultime : permettre à n'importe quel utilisateur de se connecter aux différents réseaux sociaux, de gérer son agenda ou une liste de courses.

On pourrait presque résumer le projet à un jeu de construction. Car l'équipe a dû utiliser et intégrer le matériel et le logiciel en prenant en compte des éléments extérieurs comme des services web, le réseau. Le projet a deux points d'accès pour les utilisateurs finaux, le premier est la tablette en elle-même, le deuxième le « market », magasins d'application en ligne.

Une possibilité d'ajout d'élément matériel externe est également possible comme une



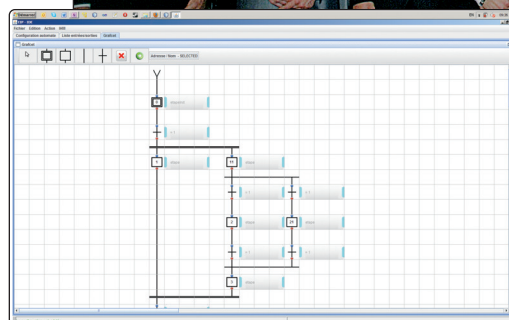
webcam ou des haut-parleurs s'ils ne sont pas de base sur la tablette.

Sur la partie technique, TWE est un ajout graphique du système. Il utilise Qt. Les widgets sont générés grâce à un « widget manager ». L'utilisateur peut aussi communiquer directement avec le « market » qui est hébergé sur un serveur. Le « market » utilise prestashop lié à une base de données (MySQL ou PostgreSQL).

Le renouveau de l'automatisme...

Voici un des projets les plus originaux ou tout du moins que l'on ne s'attendrait pas à voir à l'Epitech : IDSys. Il s'agit tout bonnement de proposer une nouvelle approche du pilotage des automates par de l'informatique « standard » ! Mais un ordinateur normal ne peut pas piloter un automate, ni contrôler les E/S, il faut des cartes spécifiques. Le projet se décompose donc en deux parties, la première consiste à réaliser un « Driver » qui gèrera la commande des Entrées/Sorties de la machine. La seconde partie du projet vise à réaliser un environnement de développement adapté aux automaticiens, puisque développé en collaboration avec eux.

Techniquement, le projet est plutôt excitant car mêlant l'industriel et l'informatique pure. Il a fallu mettre au point la partir Driver qui consiste en une machine micro ATX avec un disque SSD, donc sans pièce mécanique, le but étant d'être le plus fiable possible. Le système utilise un OpenWRT patché RTLinux. Là encore, l'équipe voulait atteindre la fiabilité maximum, seule



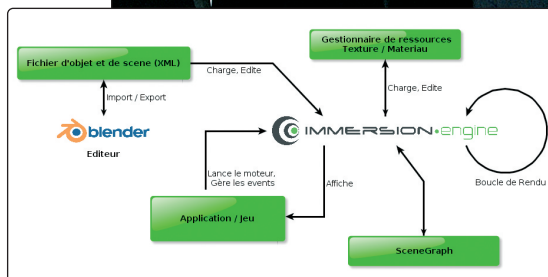
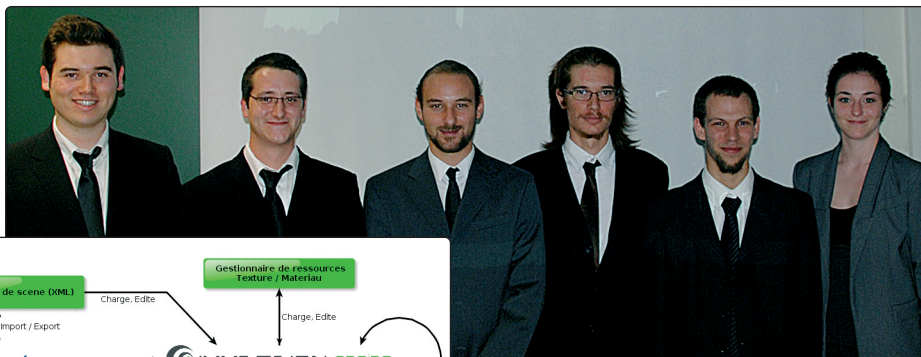
garantie dans une chaîne de fabrication où la moindre panne coûte cher ! Ce sont des milliers d'entrées/sorties qu'il faut contrôler et checker en temps réel. Le tout fut développé en langage C pour sa stabilité et ses performances. L'IDE quant à lui a été développé en JAVA, pour permettre une

portabilité plus aisée et une exportation WEB. Une des difficultés est de convaincre les automaticiens car ils sont encore réfractaires aux PC normaux, cependant, les grands acteurs du marché commencent à s'engouffrer dans cette nouvelle approche comme Schneider. Sur l'automatisme, rien ne vaut les tests réels mais l'équipe ne développe pas les machines. Il faut donc trouver des partenaires et des sociétés prêtes à tenter l'aventure.

Bravo pour ce projet, très éloigné des projets EIP habituels de l'école. Preuve que dans le domaine industriel, il y a des belles choses à faire !

Immersion engine, vers la réalité augmentée !

Parmi les nouveaux projets EIP, nous avons eu un coup de cœur pour le groupe Immersion engine. Il s'agit de créer un moteur de rendu temps réel et photoréaliste. Les deux mondes étaient jusqu'à présent à peu près hermétiques. Cette innovation permet d'obtenir un réalisme accru sur le rendu de scènes de jeux par exemple, ou pourquoi pas dans un proche avenir, dans un navigateur web, sur des tablettes pour améliorer l'expérience utilisateur. Le défi était grand et il a fallu presque 4 ans pour aboutir à un moteur fonctionnel.



La méthode

Pour arriver à cette combinaison avec des performances d'affichage du temps réel, l'équipe a utilisé la méthode du lancer de rayon (ray tracing) qui permet de restituer des propriétés optiques telles que réflexion ou transparences, à l'image des films d'animations. Mais il faut que le rendu soit interactif et s'adapte à chaque utilisateur et interaction de ce dernier. La finalité d'immersion engine est le jeu vidéo, le serious game. Pour arriver à cette performance, trois types d'optimisation ont été utilisés :

- algorithmes pour les scènes : le nombre d'objets à traiter a pu être augmenté grâce au partitionnement d'espace
- bas niveaux (caches processeurs) : optimisation processeur, notamment sur plateforme matérielle grand public
- optimisation matérielle : le calcul distribué

sur GPU est en cours de réalisation mais demande beaucoup de travail. Cette approche devrait permettre un résultat fiable sur les illuminations des scènes.

Pour arriver au photoréalisme et que le moteur soit réellement utilisable dans des projets interactifs, il faut arriver au minimum à 24 images par seconde.

Le côté technique

Le projet Immersion Engine comme vous l'aurez compris consiste à proposer un moteur réaliste fonctionnant sur le PC de tout le monde, qu'il soit sous OS X, Linux ou Windows. Surtout, les performances de rendu doivent être temps réel sur des PC standard, ce qui complique le travail des étudiants. Il fut décidé d'utiliser OpenCL et le kit de développement d'AMD. Ce choix a été fait après de

nombreux débats. Le multiplateforme a posé d'autres problèmes dans la réalisation car il fallait un code indépendant et surtout une indépendance totale par rapport aux ressources matérielles, donc trouver le bon compromis entre performance et indépendance matérielle... L'avantage d'OpenCL est tout de même sa portabilité et son fonctionnement en mode cluster. Le futur immédiat est de passer aux processeurs APU d'AMD et de terminer le calcul distribué GPU, plus loin, ce sera une version du moteur pouvant fonctionner dans un navigateur web.

Un futur plein d'ambitions !

Si nos étudiants regardent la réalité augmentée, voire la télévision connectée, le futur immédiat est de finaliser le modèle actuel et de créer une entreprise courant 2012. Un partenariat a déjà été signé avec l'INRIA, d'autres pourraient intervenir. Mais beaucoup de détails restent à régler, comme le mode de licence.

Site : <http://immersion-engine.com>

Projet bonus : Nios ou la plateforme de jeux communautaire, nouvelle génération

Nios est la reprise d'un précédent projet, Fleem. Il s'agit d'une plate-forme de téléchargement de jeux open source. Mais Nios a voulu élargir le concept initial pour y introduire un véritable réseau social, un portail communautaire pour mettre en relation développeurs, musiciens, designers, etc. Car aujourd'hui, un jeu nécessite une grande variété de compétences. Et encore faut-il les mettre en relation, fournir un cadre de travail structuré et tout le temps disponible. Bien plus qu'une aide à la gestion de projets libres, Nios a pour ambition d'offrir une structure d'échange aux passionnés ou



amateurs manquant de moyens. Une des difficultés a été de mettre en place une véritable gestion de projet pour pouvoir le faire avancer et des problèmes techniques durent être résolus. Ainsi, Rails fut remplacé par PHP, les compétences étant plus simples à trouver ! L'équipe a dû faire des choix pour avancer le plus vite possible. L'ar-

chitecture fit l'objet de débats dans l'équipe et des réunions par skype furent régulièrement organisées pour discuter, trancher les points d'architecture. Au-delà de cela, la question de l'interface fut délicate car la vision d'un développeur n'est pas celle d'un musicien ni celle d'un designer. La charte graphique fut un gros obstacle. Techniquement, Python fut utilisé (notamment parce que le projet Fleem était en Python) mais aussi parce qu'il existait plusieurs bibliothèques très utiles (PyQT, PyXmpp pour le chat, Hashlib (cryptage). Pour la partie purement web, le choix s'est arrêté à Cakephp.

Casus Belli, le MOD puissance Crysis

Plusieurs EIP tournaient autour du jeu vidéo, dont Casus Belli. Ce projet a l'ambition de créer une évolution du jeu Crysis War, en utilisant les fonctionnements FPS des jeux à l'instar de Battle Field ou Counter Strike. Parallèlement à ce travail, le but était aussi de proposer aux joueurs, à la communauté, tous les outils et templates nécessaires pour créer leurs propres variantes à Casus Belli. Il s'agit d'un MOD réaliste et multi-joueur pour Crysis Wars incluant plus



de 500 nouveaux éléments, de nouvelles images, de nouvelles musiques, 20 armes modernes, de nouveaux objectifs de jeux, modding plateforme, etc.

L'objectif était de retrouver le game play et le niveau fonctionnel des jeux FPS des dernières années, dicit l'équipe. Pour les hardcore gamers, le choix du moteur graphique Crysis Wars n'est donc pas le fruit du hasard, bien au contraire. Le SDK du moteur était gratuit mais les possibilités offertes étaient parfois limitées, comme avoir une projection d'un corps après une explosion ou la possibilité de relever un blessé. Mais de l'aveu même des étudiants, le développement d'un jeu demande aujourd'hui des compétences très diverses : du développement pur mais aussi des spécialistes en graphisme, design, audio, animation. Pour compléter l'équipe, Casus Belli travaille avec une association d'artistes : French Touch Studio, très actif au sein de la communauté Crysis. Pour l'anecdote, c'est French Touch Studio qui créa le projet. Un long travail a dû être fait sur le moteur, les possibilités du moteur, la documentation disponible étant très limitée, voire inexistante. Un premier prototype fut alors développé avant de pouvoir rentrer dans la 2e phase du projet : mise en production à partir de septembre 2010, bénéficier d'une première version

stabilisée au début de l'été 2011 puis d'une bêta privée à partir de septembre dernier et enfin depuis fin novembre dernier, d'une bêta publique. Un important travail fut réalisé pour refaire l'interface de travail.

Architecture

L'équipe utilise le SDK 1.5 de Crysis Wars (CryEngine 2) pour développer Casus Belli. Ce SDK fournit :

- des outils pour créer des levels et autres éléments artistiques
- du code C++ et LUA pour modifier les règles du jeu

Nous pouvons voir le SDK comme un iceberg. Le code du moteur du jeu n'est pas fourni (partie immergée), en revanche nous avons accès à une API qui nous permet d'altérer suffisamment de choses pour changer une très grande partie des règles du jeu de base (partie visible).

Pour créer Casus Belli, l'équipe a utilisé :

- du C++ pour modifier les éléments communs aux différents modes de jeux
- du LUA pour les entités et modes de jeu
- du Flash/Actionscript 2.0 pour l'interface utilisateur.

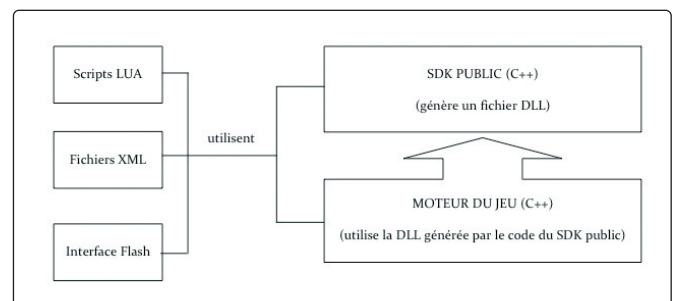
La compilation du code C++ génère une DLL qui sera ensuite chargée par Crysis.exe. Le code LUA en revanche n'a pas besoin

d'être compilé. La partie Flash était basée sur une ancienne version de la technologie. Définir un modèle de code fut un des grands défis du projet car si ce modèle permet de standardiser le développement, il apporte aussi des contraintes et rapidement, il a fallu mettre en place une véritable gestion de projet. La motivation ne fait pas tout et cela peut rapidement partir dans tous les sens. Des séances régulières de tests étaient organisées pour traquer les bugs, ajuster le game play, l'interface.

L'avenir ?

La communauté et le projet devraient continuer. Mais l'avenir dépendra de la communauté et du succès du jeu. Une partie du code sera ouvert, hormis la partie C++ (pour le moment). Il est question de passer à la v2 du moteur graphique mais la migration s'avèrerait compliquée. Et si les développeurs - étudiants se passionnent pour Casus Belli, le jeu n'est pas le domaine le meilleur, car la concurrence est vive entre développeurs, mais l'expérience apportée, le fait d'avoir sorti une version stable sont des « plus » indéniables pour les étudiants !

Site officiel : <http://www.casusbelli-mod.com/main/>



D.R.

Le retour du développement natif en C++

Évoquer le développement natif, cela implique qu'une application appelle en direct les primitives (ou API) du système d'exploitation sur lequel il s'exécute. Or, les systèmes d'exploitation sont majoritairement faits en C/C++. Le développement natif se traduit naturellement en développement C/C++ car les primitives des systèmes sont soit des fonctions, des classes ou des interfaces exposées pour être utilisées par un compilateur C++.

C++ est dans l'O.S et pas seulement dans Windows, il est dans le navigateur web et pas seulement dans Internet Explorer. La communauté C++ ne se recrute pas seulement au travers des universités et des entreprises, c'est la plateforme la plus populaire (25%) devant Java (17%), C# (7%), PHP (6%) et Objective-C (6%) – cf. l'indice TIOBE disponible sur <http://www.tiobe.com>. C++ veut dire développement natif. C++ permet d'accéder à toutes les fonctionnalités sous-jacentes du système d'exploitation. C'est une caractéristique unique.

Ce n'est pas le cas des langages managés comme C# ou Java qui sont dépendants d'un framework et d'un runtime d'exécution. Et lorsqu'une fonctionnalité n'est pas exposée au travers du couple framework/runtime, un langage managé doit faire un appel à des API natives au travers d'un système de traversée de couches (marshalling) souvent exotique à coder (P/Invoke en C#, JNI en Java).

La nouvelle génération d'application appelle le développement natif. La volonté de faire des interfaces utilisateur comme dans le film « Minority Report » appelées aussi « Native UI » en est le meilleur exemple. L'autre argument est la puissance et la performance qu'apporte C++ et il n'y a pas d'équivalent. Un très haut degré d'abstraction mais sans compromis sur la performance.

Aujourd'hui, l'essor des plateformes mobiles et des tablettes et des périphériques portables nécessite l'utilisation d'outils natifs pour travailler avec le socle hardware et le système d'exploitation du périphérique. Les batteries sont limitées et les préserver est une priorité.

Comment cela se traduit-il ? Le programme doit gérer tout de A à Z. Et pour cela, C++ est le bon langage car rien n'est caché, on a accès à tout.

“ C++ permet d'accéder à toutes les fonctionnalités sous-jacentes du système d'exploitation. ”

> Les évolutions de C++

C++ a évolué au travers de différentes étapes de standardisation, C++98, C++03, TR1, C++0X et maintenant C++11, la dernière version sortie en mars 2011. L'exemple le plus intéressant dans ce nouveau C++ est le rapport à la gestion du cycle de vie des objets. Avant, le code C++ utilisait des appels à new et le code pouvait produire des fuites mémoires difficiles à reproduire et donc à corriger. Avec les templates comme `shared_ptr<T>` et `weak_ptr<T>`, il est possible de s'affranchir de certains problèmes liés à la restitution de la mémoire au travers d'un mécanisme de compteurs de références sophistiqué et d'une implémentation qui ressemble au pattern Observateur. Bjarne Stroustrup, inventeur du C++ déclare entre autres, que « *C++ is the best language for garbage collection, principally because it creates less garbage* ». C++ est le meilleur langage pour le garbage collector car il ne crée rien à nettoyer. La notion de GC n'existe pas en C++. Concernant la notion de garbage collector disponible dans les autres langages, cela ne marche pas en C++. C++ supporte la diversité. Il y a plusieurs systèmes de gestion mémoire et pas un moyen unique. Les allocations peuvent se faire en tirant parti soit du langage (`new()`, `delete()`), soit de la librairie standard STL (utilisation des templates `shared_ptr<T>`) ou bien de l'OS (exemple `VirtualAlloc()` en Win32). Le

problème n'est pas d'avoir une application qui libère automatiquement la mémoire mais plutôt d'avoir une application qui démarre vite et qui soit rapide. Dans l'industrie du jeu vidéo, les applications pré-allouent un immense bloc de mémoire, le partitionnent et fournissent plusieurs mécanismes de caches. Dans le milieu de l'embarqué, certains programmes doivent allouer la mémoire au début et doivent être en consommation constante et ne plus allouer de mémoire supplémentaire. Seul C++ peut garantir cela. Les GC sont fantastiques pour les langages artificiels et les langages managés, pas pour C++. En C++, les pointeurs partagés permettent de gérer le cycle de vie des objets efficacement. Dans le monde .NET, il n'existe qu'une seule méthode pour allouer de la mémoire. Pour être rapide et efficace, utiliser un GC n'est pas une bonne option.

Le responsable de Visual C++ (Ale Contenti) préfère cette maxime « *C++ brings Power and Performance without compromise* ». C++ ne sacrifie jamais le pouvoir donné au programmeur en faisant des compromis sur les performances.

> C++ Renaissance

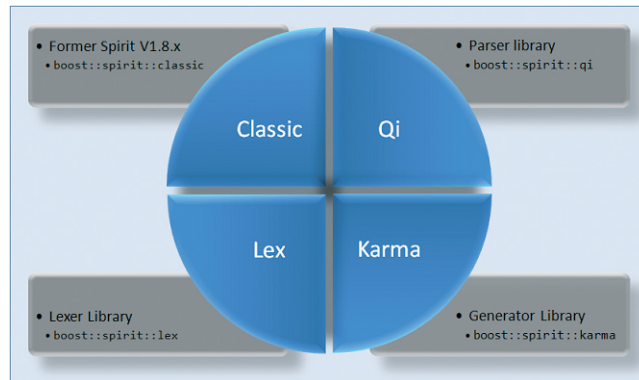
Les conditions sont réunies en 2011 pour que C++ revienne au centre de l'actualité. Le nouveau C++11 vient d'être standardisé. Il y a de plus en plus de personnes qui posent des questions et qui veulent revenir à des fondamentaux comme les applications rapides et fluides. C++ est utilisé dans différentes lignes métier comme les mobiles, les serveurs, les jeux, partout. La communauté n'est pas la plus grande mais elle continue de grossir.

Même chez Microsoft, là où l'engouement pour le code managé est très grand, tous les produits majeurs sont faits en C/C++ : Windows, Office, SQL Server. C++ est à la fois un langage de très bas niveau mais qui

peut aussi être très abstrait. Microsoft a commencé par faire du C avec des classes et maintenant pratique le C++ moderne. Comme l'électricité, son utilisation est naturelle et il est construit de telle manière que le pouvoir et la performance sont fournis par défaut. Il permet de faire des choses rapides comme en C mais aussi des opérations typées, de la programmation objet puissante. La performance est impressionnante. Il suffit de jeter un coup d'œil au code assembleur généré qui est fantastique. Exemple : les compilateurs et optimiseurs de code C++ utilisent les registres CPU. En C++, la performance n'est pas négociable. De plus, les optimiseurs de code savent parler aux architectures processeurs pour injecter les bonnes instructions et les bonnes extensions CPU. En termes d'avantage marketing, le support aux dernières innovations ou le support des derniers matériels est presque toujours fourni sous forme d'accès natif. C++ permet cela. Concernant la productivité, celle-ci est différente en natif et en code managé. Pour la conception logicielle, ce n'est pas le dernier framework ou la dernière librairie à la mode qui va orienter la conception d'une application. En C++, vous n'êtes pas obligés de faire de l'orienté objet ou des classes. On peut utiliser directement des fonctions. De plus, C++ permet de faire de la programmation à base de meta templates (templates meta programming). La syntaxe est un peu exotique mais c'est toujours du C++, et lui seul permettra cela. La librairie Boost::Spirit est souvent citée en exemple. Faites un tour sur www.boost.org.

> Les nouveaux challenges parallèles

Maintenant, le challenge qui se profile à l'horizon est le passage à la programmation multi-core et l'exploitation pleine et entière des nouvelles architectures processeurs. Dans ce domaine, Intel travaille en assembleur mais son principal support est le langage C++. Dans les prochaines années, nous allons avoir des librairies construites à base de templates avec un très haut degré d'abstraction mais une implémentation qui va directement tirer parti des dernières extensions des processeurs et aussi des cartes graphiques (GPU). Microsoft fournit dans Visual C++ la librairie PPL (via la runtime C) qui est construite à base de classes, d'interfaces et de templates. Intel propose TBB. Dans son prochain Visual Studio,



Microsoft va fournir une librairie AMP pour tirer parti des GPU en utilisant les templates, DirectX et les pixels shaders. C++ est plus qu'un langage. Il est indissociable des librairies, la runtime C, la runtime C++ nommée STL.

Une librairie peut être générique et portable comme une librairie de calcul. Une interface utilisateur peut être portable. Les développeurs C++ sont assez sophistiqués pour comprendre ce qui peut être portable et ce qui doit être dépendant d'une plateforme spécifique. Le « one size fits all in C++ » n'existe pas. Le langage C++ a tellement de fonctionnalités élégantes comme les templates et des moins élégantes comme les macros. Le langage possède des attributs uniques et C++ est associé à l'excellence dans le domaine du développement logiciel. Si vous êtes un développeur C++, vous êtes forcément un peu plus curieux. C++ est natif et les systèmes d'exploitation aussi, donc il est facile de faire des applications qui exploitent plus et sont plus performantes. C++ est connu comme un langage dangereux et non sécurisé : c'est un mythe et en plus, c'est faux. Il s'agit d'un argument marketing pour vous détourner vers d'autres langages ! Les outils de développement comme Visual Studio vont contenir de plus en plus de librairies et spécificités dédiées uniquement à C++. Rendez-vous avec Visual Studio 2011.

> Pourquoi utiliser C++ ?

Pourquoi utilisons-nous C++ demande Herb Sutter, qui préside le comité de standardisation C++. Il y a 3 éléments de réponse :

- la performance par rapport au coût

- la performance par rapport au nombre de transistors disponibles (CPU, GPU)

- la performance par rapport aux cycles CPU
- Dans les années 90, C++ est devenu un langage standard. Dans les années 2000, la productivité est le sujet majeur et ce fût l'âge java, C#,

les langages managés. Mais la question est : peut-on tout faire avec un langage managé ? La réponse est non. Lorsque Herb Sutter a rejoint Microsoft, le futur système d'exploitation, nom de code Longhorn étant en phase de développement, les nouvelles API seraient managées. C'était l'idée.

Microsoft a essayé d'introduire du code managé dans le système d'exploitation et ce fût un challenge intéressant. Mais le résultat n'était pas bon et loin d'être parfait, que ce soit sur la robustesse des interfaces et du code développé, les performances et le mode de déploiement. C'est la raison pour laquelle Windows est toujours en C et C++. C# est fait pour la productivité. Depuis la dernière décennie, les développeurs se sont tournés vers les langages managés. Par contre, les utilisateurs demandent des interfaces fluides, des tablettes, des périphériques portables évolués, des caméras temps réel, des fonctionnalités d'OCR temps réel et des expériences utilisateurs qui répondent vite. Aujourd'hui, seul le langage C++ permet de répondre à cela. Tout est histoire de performance. Les langages qui sont construits autour de l'optimisation de code auront une nouvelle vie, disait Herb Sutter en 2004. Les conditions sont aujourd'hui réunies en 2011.



Christophe Pichaud

.NET Rangers by Sogeti

Consultant sur les technologies Microsoft

christophepichaud@hotmail.com

www.windowscopy.net

Vidéos Channel9 :

<http://channel9.msdn.com/Shows/C9-GoingNative/GoingNative-0-Help-us-fly-this-plane-Some-modern-C-Meet-Ale-Contenti>

<http://channel9.msdn.com/Shows/Going+Deep/Craig-Symonds-and-Mohsen-Agsen-C-Renaissance>

<http://channel9.msdn.com/posts/C-and-Beyond-2011-Herb-Sutter-Why-C>

<http://channel9.msdn.com/Shows/Going+Deep/Mohsen-Agsen-C-Today-and-Tomorrow>

<http://channel9.msdn.com/tags/STL/>

Notre sélection de livres

COLLECTIF

Sécurité PHP 5 et MySQL

Eyrrolles, 35 €



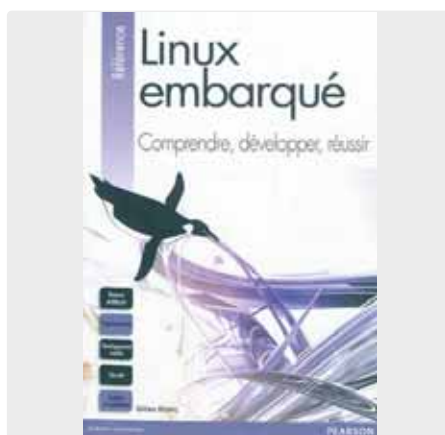
La sécurité demeure une des priorités du développeur web. Pour mettre en place une politique de sécurité à la fois efficace et souple, sans être envahissante, il est essentiel de maîtriser les nombreux aspects qui entrent en jeu dans la sécurité en ligne : la nature du réseau, les clients HTML, les serveurs web, les plateformes de développement, les bases de données... Autant de composants susceptibles d'être la cible d'une attaque spécifique à tout moment.

GILLES BLANC

Linux embarqué

Pearson, 43 €

Découvrez un monde rarement connu du public et même des développeurs : le système embarqué. Et pourtant, sans l'embarqué : pas de téléphone, pas de caisse enregistreuse. Ce livre est un excellent condensé des problèmes. Chaque aspect de réalisation d'un projet Linux embarqué est abordé : étapes préparatoires et aspects économiques, juridiques, décisionnels/managériaux en jeu ; rédaction des spécifications : réflexion sur l'architecture, rédaction des



spécifications, mise en place de l'environnement de développement ; réalisation technique : pas de tutoriels, mais un choix d'outils présentés en détail avec une grille de lecture permettant au développeur d'adopter la solution idéale. On pointe ensuite pour chaque outil vers un tutoriel en ligne existant et dont on sait qu'il sera actualisé.

RONAN CHARDONNEAU

Google Analytics

Eni, 26,50 €



Maîtriser son site web, connaître les statistiques, réagir en conséquence, mener des campagnes, comprendre la fréquentation, être webmaster est devenu un métier complexe et stressant. Mais avec les bons outils, la bonne analyse, il est possible de faire beaucoup. Google Analytics est un outil incontournable pour les statistiques web mais pas toujours simple à maîtriser. Cet ouvrage a l'ambition de vous faire découvrir les subtilités d'Analytics, les fonctions et comment le mettre en œuvre chaque jour en utilisant les chiffres clés. Indispensable !

COLLECTIF

Programmation en Python pour les mathématiques

Dunod, à paraître

Multiplateforme, puissant, souple, Python est devenu un des langages les plus connus même s'il n'est pas le plus utilisé. Il est particulièrement apprécié des pédagogues qui y trouvent un langage où la syntaxe, séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation. C'est le langage le plus répandu dans le monde scientifique. Les auteurs présentent tout ce que le mathématicien doit savoir sur ce langage : comment se le procurer, comment l'installer, l'utiliser, quelles sont ses fonctionnalités proprement mathématiques et quels problèmes sont susceptibles d'être résolus grâce à lui.

Java User Group de Grenoble alias AlpesJUG

A Grenoble, les amateurs de Java peuvent approfondir leurs connaissances grâce à AlpesJUG. Chaque mois, des conférences sont organisées autour de nombreux thèmes : Git, NoSQL, DevOps, DataGrid, GridGain, Scala... Après la réunion les participants se retrouvent

pour un échange informel autour d'un verre. Le 15 décembre dernier, la réunion avait pour thème « datagrid et distributed cache avec InfiniSpan de Jboss, avec comme intervenant Galder Zamarreno de l'équipe JBoss ! Le groupe cherche de nouveaux sponsors / partenaires en plus de www.open-groupe.com

Les liens du JUG :

Site Web : <http://www.alpesjug.fr/>

Agenda :

<https://www.google.com/calendar/ical/3eaph2fdu3lr1equpn6la5jqdk%40group.calendar.google.com/public/basic.ics>

Twitter : <https://twitter.com/#!/alpesjug>

Forum :

<https://groups.google.com/forum/#!/forum/alpes-jug-annonces>

LinkedIn :

<http://www.linkedin.com/groups?gid=2350443>


Avril 2012

Devoxx arrive en France !



Voilà une des annonces les plus fortes de cette fin d'année ! Le grand événement belge Devoxx dédié au monde Java débarque en France du 18 au 20 avril prochain ! C'est une nouvelle importante car depuis plusieurs années, nous n'avions plus grand chose à nous mettre sous nos codes Java. Les Java Days n'existaient plus et Oracle n'organise pas d'événements à la hauteur. Devoxx constitue une force incomparable : 3 300 participants en novembre dernier en Belgique. Devoxx France travaille activement avec Paris JUG pour proposer un événement à la hauteur et excitant : 3 jours de conférences (75 % français, 25 % anglais). La conférence vise aussi bien le développeur que l'architecte et les décideurs. La conférence se déroule du mercredi 18 au vendredi 20 avril 2012 au centre de conférence de

l'hôtel Marriott Paris Rive-Gauche. Pour la première année, l'équipe d'organisation vise 700 personnes. Un espace d'exposition de 660 m2 est prévu pour les partenaires de l'événement. 4 salles de conférence de 270 places permettent de suivre des présentations d'une heure sur 4 thèmes majeurs :

- web, mobile et cloud
- les langages non java dans la JVM
- entreprises et pratiques
- Java / JEE et architectures.

La journée du mercredi reprend le format Devoxx : sessions de 3 heures pour une véritable formation en alternance, des sessions courtes pour présenter des outils et logiciels. Une grande plénière sera organisée pour faire le point sur Java, les perspectives, les technologies. Attention, Devoxx France est payante : 450 € H.T. pour les trois jours.

Si vous assistez uniquement aux journées du jeudi et vendredi, 350 € H.T.

Bloquez impérativement votre agenda !



agenda

➤ JANVIER

12 janvier 2012, Talence,
19 heures, salle TD6 au premier étage de l'ENSEIRB-MATMECA.

IHM et ergonomie, organisé par les Tontons Flexeurs.

Comme d'habitude c'est gratuit et ouvert à tous, alors inscrivez-vous vite sur

<http://ttfx6.eventbrite.com/>

Du 27 au 29 janvier 2012,
Salle Trianon, place du 18 juin 1940
14780 Lion sur Mer

Rencontres du livre 2012 (2e édition).

http://www.rencontres-du-livre.org/deuxiemes_rencontres_du_livre_lion_sur_mer/index.html



➤ FEVRIER

Du 7 au 9 février, paris 17e Palais des Congrès, les **Microsoft TechDays**, le rendez-vous incontournable des développeurs professionnels

<http://www.microsoft.com/france/mstechdays/>

➤ MARS

Du 20 au 21 mars 2012, La Maison Internationale, 17 boulevard Jourdan, 75014 Paris **Congrès Big Data Paris 2012**. Ce sera le premier événement de cette ampleur à décrypter le phénomène numérique Big Data, annoncé comme la prochaine révolution de l'informatique mondiale. <http://www.bigdataparis.com>

Notre sélection web du mois

➤ Que se passe-t-il réellement chez **Palm** ?

Le blog officiel donne parfois des informations intéressantes comme l'annonce de la disponibilité en open source du système. A découvrir et à lire ici :

<http://developer.palm.com/blog/>

➤ Où trouver les dernières informations sur le monde Java et sa communauté,

java.blogs propose une agrégation de différents blogs, à consulter régulièrement pour connaître les annonces, événements, les bugs, etc. :

<http://javablogs.com/Welcom.action>

➤ En Français, un autre blog **Java** qui pourra vous servir : <http://blog.alexishassler.com/>

➤ Une **petite histoire** sympathique des langages de programmation, en Français :

<http://www.scriptol.fr/programmation/histoire-langages.php>

➤ **Facebook** tient aussi un blog spécial développeur : <http://developers.facebook.com/blog/>

Avant le 20 février 2012 : Innovact Campus Awards



Pour la 10e année, les Innovact Campus Awards défient les étudiants, jeunes chercheurs, entrepreneurs de toute l'Europe pour montrer, démontrer leurs projets, leur innovation ! Les Innovact Campus Awards 2012 sont ouverts à tous les étudiants et jeunes chercheurs européens, dans le cadre ou non de leur formation. Les projets innovants proposés peuvent être en phase de conception, en cours de réalisation ou déjà lancés. Le jury, constitué de professionnels venus de toute l'Europe - chefs d'entreprise, journalistes, scientifiques - distinguera les candidats qui ont fait preuve d'imagination et d'innovation dans leur démarche de création, quel que soit le produit ou l'activité. Le dossier doit être déposé avant le 20 février 2012. Téléchargez le dossier de candidature sur www.innovact.com ou www.verbatim-communication.fr

II Appel aux groupes utilisateurs !

Programmez! s'ouvre à toutes les communautés. Notre volonté est de se rapprocher des communautés et groupes utilisateurs. Chaque mois, Programmez! publiera l'agenda, les événements, les comptes-rendus des groupes utilisateurs. Mais sans vous, nous ne pouvons rien faire ! Envoyez-nous vos agendas, vos événements, vos propositions de comptes-rendus et d'articles ! Programmez! est le media idéal pour les communautés et les groupes utilisateurs! - contact : ftonic@programmez.com

« Depuis mes dix ans je suis passionné par l'informatique »

Nous oublions souvent que derrière le logiciel, la techno, il y a le développeur. Chaque mois, nous allons en rencontrer une ou un qui va nous dévoiler son quotidien, son travail, sa passion. Ce mois-ci, Julien Dollon nous parle de son parcours, de sa passion et de l'ambiance à San Francisco, sa nouvelle résidence depuis quelques mois.



Ce n'est pas Live at Pompei (les amateurs apprécieront sans aucun doute) mais Live at Silicon Valley !

Comment es-tu tombé dans l'informatique et plus spécialement dans le développement ?

Julien : Depuis mes dix ans je suis passionné par l'informatique. Comme beaucoup de petits jeunes je me suis de suite orienté vers le "hacking" (oui, car à 10 ans on est persuadé d'être le meilleur du monde, à 24 aussi). J'ai commencé par le scripting sous mIRC pour faire des "attaques de clones" ou des "smurfs", j'avais déjà une certaine prédilection à « embêter mon monde ». Puis j'ai évolué jusqu'à mon bac vers de la vraie sécurité : j'ai étudié toute la certification Cisco CCNA, développé mes buffers overflows en C et mes shellcodes en ASM, étudié une foule de protocoles réseaux, de distributions Linux et BSD du moment et j'étais un pro du PHP3. C'est d'ailleurs une très bonne chose de m'être tourné aussi longtemps vers le monde libre, ce qui me permet d'avoir un

peu de recul aujourd'hui, même si je suis plus orienté technos Microsoft.

Pour toi, qu'est-ce qui fait que l'on aime toujours et encore le développement, la technique ?

Julien : Quand j'étais enfant, je me souviens de la joie que j'avais quand je découvrais un nouveau jouet. Le cœur s'accélère, on veut tout de suite l'essayer, le connaître, qu'il n'ait plus aucun secret pour nous. On imagine tout ce qu'on va pouvoir faire avec ce jouet (et si en plus on peut le montrer aux copains pour crâner, c'est encore mieux). Remplacez le mot jouet par « nouvelle techno » et vous vous retrouvez avec le même sentiment. Bon on peut aussi remplacer jouet par bien d'autres choses. La technique, une fois qu'on tombe dedans, nous habite, nous motive ! Elle permet d'atténuer notre soif d'apprendre.

Tu as gardé un regard très geek : gadget, veille techno. C'est important pour ton job et ta passion ?

Julien : A ce propos, tu n'aurais pas des goodies Programmez !? (ndlr : nous allons y réfléchir)... Il est difficile de trouver le BON goodies/gadget, pas le truc basique, type balle rebondissante, mais plutôt le goodies de compétition : une sacoche de PC Windows Azure, un t-shirt I Love Windows Phone ou encore une casquette Tech Ed 2012. Bon après, on est content même si le t-shirt est un XXL alors que je fais du M. Côté veille techno, cela m'a permis de me démarquer dans ma carrière. Le fait de toujours se tenir au courant, d'être en constante recherche (livres, blogs) va permettre de

développer une grosse culture sur la technologie en question. Depuis 5 ans, il ne se passe pas un jour sans que je prenne environ 30 minutes pour lire tous les blogs (développez, programmez, Dotnet-France...) ainsi que les groupes facebook spécialisés. Dans un second temps, j'essaie de lire un livre par mois, de préférence en anglais car je trouve le niveau bien meilleur. Résultat, à force de lire, on acquiert des best practices et une expérience à travers les yeux des autres experts qui partagent. J'avoue cependant que je concentre/limite ma veille technologique au monde de l'entreprise dans laquelle j'évolue. Dès que cela touche des choses comme les jeux vidéo, un autre langage, etc, je ne m'y aventure pas. Il est déjà dur de maîtriser une technologie.

Etre développeur n'est pas toujours facile : pression, évolution constante, frustration des projets et des "chefs", c'est quoi pour toi être développeur aujourd'hui ? Le job a-t-il changé depuis tes débuts ?

Julien : N'étant pas très vieux, je ne peux pas trop comparer avec mes débuts. La seule chose que je remarque c'est que non, il n'y a pas tant de pression que ça et pas tant de frustrations.

Du moins, ici, à San Francisco. Contrairement à la France, ici on n'a pas cette culture du « petit chef ». Mon épouse est aussi développeur .NET et dans les deux entreprises où nous sommes on ressent bien l'importance du développeur. En France, j'ai eu la chance d'enseigner dans une école spécialisée en informatique. Mon test préféré en début de cours sur la conception de logiciel est de demander « Qui veut être chef de projet plus

tard ? ». La réponse est quasi unanime, hormis peut-être les deux barbus de devant, et les trois « délinquants » du fond, la totalité veut devenir CHEF de projet! Quoi qu'il en soit, on a un problème en France, on pense (et ça reste vrai) que pour évoluer socialement (et donc indirectement financièrement) il faut nécessairement passer par le management. Il faut être fier d'être développeur, ce sont des créateurs. Aux USA on les appelle les rockstars dans certaines entreprises. Les dernières statistiques à San Francisco montrent d'ailleurs qu'un chef de projet gagne autant qu'un développeur moyen. Un chef de projet ça se remplace rapidement, un bon développeur c'est une autre histoire !

Et en dehors du boulot, qu'est-ce que tu aimes faire ? Comment trouves-tu l'équilibre entre travail, vie privée, passion, famille ?

Julien : J'ai la chance d'avoir une femme passionnée de développement comme moi ! Donc quand je rentre du travail avec la dernière tablette Windows 8, nous allons passer la soirée à jouer avec.

Le souci est de savoir dire « stop » sur le temps que je passe sur le PC. Etre développeur et entrepreneur est très consommateur en temps.

Depuis 1 an, j'essaie de limiter mes projets de développement à côté du travail qui nécessitent beaucoup d'investissement, je préfère me concentrer sur des projets communautaires comme l'écriture de mon blog julien.dollon.net, l'écriture de livres ou l'ani-

mation d'une webtv (rdvexperts.com). Je ne suis pas vraiment un modèle d'intégration sociale alors je vais éviter de donner mes super conseils .

Tu as fait de jolis parcours au concours ImagineCup de Microsoft, tu es maintenant dans la Silicon Valley. Le défi est-il ta seconde nature ? Comment ressens-tu le fait d'être développeur en France et aux US ?

Julien : En entendant la question, j'ai eu l'impression d'être lady Gaga, super star, invitée sur TF1 (ndrl : difficile de faire toujours original dans les questions, nous ferons mieux la prochaine fois). Le défi est ma première nature, je le vois partout et je ne vis que pour ça. Il est difficile pour moi de comparer le métier de développeur en France et aux USA car je suis passé d'un monde de service à un monde produit. La concurrence ici n'a pas vraiment l'air d'être rude, il y a un gros manque de développeurs. Dans mon entreprise, celle de ma femme (et Facebook) il y a plus d'étrangers que d'américains.

Peux-tu nous présenter ton quotidien en quelques mots ?

Dodo – Code – Sushi, oui car les pizzas c'est bien au début, mais je dois faire attention à ma ligne si je veux continuer à rentrer dans mon teeshirt Silverlight. J'ai essayé de demander à Microsoft un nouveau teeshirt Silverlight en L, j'ai reçu un teeshirt HTML5/JavaScript .

Des conseils aux étudiants et dévs qui te lisent ?

Julien : Pour les étudiants, le conseil que je peux donner est de se spécialiser dans une techno après la deuxième année d'étude. En plus de l'école, il faut pouvoir pendant trois ans se donner à fond : livres, blog, concours, webcast, certifications, stages. Ne pas hésiter à se faire des contacts auprès des experts lors de conférences techniques. Cela vous permettra d'avoir une bonne vue de l'écosystème dans lequel vous aller évoluer en sortant de l'école. Pour les développeurs, plus particulièrement Microsoft, le conseil que je vous donne : foncez apprendre HTML5 !!! Une petite citation pour la fin qui m'a bien fait rire :

« Aujourd'hui, la programmation est devenue une course entre le développeur, qui s'efforce de produire de meilleures applications à l'épreuve des imbéciles et l'univers, qui s'efforce de produire de meilleurs imbéciles. Pour l'instant, l'univers a une bonne longueur d'avance » Rich Cook.

Propos recueillis par François Tonic

Les adresses de Julien :

<http://www.linkedin.com/in/juliendollon>

<http://twitter.com/juliendollon>

<http://www.facebook.com/julien.dollon>

<http://gplus.to/juliendollon>



Chaque mois, nous faisons parler un développeur. Si tu veux participer, une seule adresse mail à retenir : ftonic@programmez.com



Julien Dollon avec Steve Ballmer, CEO de Microsoft Corp., à l'issue du concours Windows Phone 7

J'installe Android SDK 4.0

Avec l'explosion du marché des smartphones, et donc des markets d'applications en ligne, le marché des applications est en plein boom. De plus en plus de personnes, se mettent à développer, même celles n'ayant jamais codé pour des mobiles.

Si vous lisez ces lignes, vous faites très certainement partie de ceux qui souhaitent utiliser au maximum les possibilités de leur smartphone (ou de leur tablette), voire même d'en faire ensuite profiter la communauté.

Nous nous proposons donc ici d'installer un environnement complet de développement d'applications Android à partir de rien, et nous détaillerons le cheminement sur les trois plateformes majeures : Windows, OS X et Linux.

Note : *il vous faudra la dernière version de Java sur votre ordinateur, mais elle est en général installée sur la plupart des machines. Si ce n'est pas le cas, rendez-vous sur <http://java.com/fr/download/manual.jsp>, téléchargez et installez la version correspondant à votre système d'exploitation.*

> Eclipse

La première chose à faire est de mettre en place notre IDE (Integrated Development Environment). Un IDE est un éditeur de texte qui inclut un certain nombre d'outils qui aident au développement (tels qu'un compilateur intégré, un débogueur, un correcteur de syntaxe, l'auto-complétion du code etc.).

Note : *Savoir utiliser son IDE est quelque chose d'important, donc n'hésitez pas à naviguer dans les différentes options, préférences, raccourcis claviers et autres modules qui peuvent grandement vous simplifier la vie.*

Nous choisirons ici Eclipse, pour diverses raisons : il est disponible sur les trois plateformes qui nous intéressent, il possède une bonne intégration du langage Java et Google fournit un ensemble d'outils plugins à greffer sur Eclipse pour aider au développement d'applications. Il en existe évidemment d'autres (ex : NetBeans).

Utilisateurs Windows et Mac, rendez-vous sur le site officiel d'Eclipse (<http://www.eclipse.org/downloads>) et téléchargez la dernière version de « Eclipse IDE For Java Developers ». Installez-le comme n'importe quel autre logiciel. Si vous êtes sous Linux, deux choix : utiliser votre gestionnaire de paquets (ex :

« `sudo apt-get install eclipse` », « `pacman -S eclipse` »), ou bien télécharger depuis le site officiel d'Eclipse (<http://www.eclipse.org/downloads>), extraire l'archive et lancer le binaire « eclipse » qu'elle contient.

> SDK

Nous possédons maintenant un éditeur. Procédons à l'installation de l'outil de développement Android mis à disposition par Google : le SDK Android (Software Development Kit).

Direction donc <http://developer.android.com/sdk>. Choisissez la version correspondante à votre système [Fig.1]. Sous Windows, préférez la version en .exe, et installez-la simplement (si jamais vous changez le répertoire cible de l'installation, n'oubliez pas où vous l'avez mis). Pour Linux et OS X, téléchargez l'archive et extrayez-la quelque part dans votre arborescence en notant son emplacement. Il faudra ensuite ajouter le chemin vers cet emplacement dans votre \$PATH (variable qui indique au système où se trouvent les binaires qu'il peut utiliser). Pour cela, ouvrez un terminal et tapez « `echo 'PATH=${PATH}:/chemin de votre dossier SDK' >> ~/.bashrc` » et fermez le terminal

Exemple : si votre dossier « platform-tools » se trouve dans « ~/download/android » sur votre PC, il faudra taper : « `echo 'PATH=${PATH}:/download/android/platform-tools' >> ~/.bashrc` » [Fig.2].

> Plugin

Nous disposons maintenant du SDK Android. Il ne sert pas à grand-chose tout seul (si le cœur vous en dit, vous pouvez utiliser le binaire « adb » en ligne de commandes, pour lister les appareils branchés à votre ordinateur, ou encore ouvrir un terminal sur un de ceux-ci). Intégrons donc le SDK à notre éditeur. Pour cela, Google propose un plugin pour Eclipse : l'ADT (Android Development Toolkit).

Lancez donc votre Eclipse. Pour le premier lancement, vous aurez sans doute à choisir un « Workspace », qui correspond simplement à l'endroit où seront stockés vos fichiers sources sur votre ordinateur.

Dans l'onglet « Help », choisissez « Install New Software ». C'est à cet endroit que l'on déclare les plugins que l'on souhaite installer sur notre Eclipse. De nombreux plugins sont disponibles (comme des plugins de thèmes de colorations par exemple), mais ici nous allons simplement ajouter le plugin ADT. Cliquez sur « Add... ».

Le principe est d'ajouter une URL qui correspond à l'endroit où les sources du plugin sont mises à disposition par le développeur. De cette manière, le plugin peut facilement être mis à jour. Le nom n'est là que pour vous permettre de vous y retrouver dans vos listes de plugins.

Dans « Name », entrez « ADT Android », et dans « Location » entrez « <https://dl->



Download the Android SDK

Fig.1

Welcome Developers! If you are new to the Android SDK, please read the steps below, for an overview of how to set up the SDK. If you're already using the Android SDK, you should update to the latest tools or platform using the *Android SDK and AVD Manager*.

Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r15-windows.zip	33895447 bytes	cc2aadf7120d12b574981461736a96e9
	installer_r15-windows.exe (Recommended)	33902520 bytes	ee8481cb86a6646a4d963d5142902c5c
Mac OS X (intel)	android-sdk_r15-macosx.zip	30469921 bytes	03d2cdd3565771e8c7a438f1c40cc8a5
Linux (i386)	android-sdk_r15-linux.tgz	26124434 bytes	f529681fd1eda11c6e1e1d44b42c1432

```
[mogmi@vm03] ~ - [~]
[$] >> echo "export PATH=${PATH}:/download/android/platform-tools" >> ~/.bashrc
```

Fig.2

ssl.google.com/android/eclipse » puis validez. Dans le menu déroulant, sélectionnez « ADT Android » que vous venez d'ajouter, puis cochez la case « Developer Tools » qui apparaît juste en dessous [Fig.3].

Cliquez sur « Next » à deux reprises (confirmations des paquets qui seront installés sur votre système) et validez les conditions d'utilisation. Cliquez enfin sur « Finish » pour lancer l'installation.

Relancez Eclipse dès la fin de l'installation.

Il faut maintenant configurer l'ADT. Allez dans « Windows » puis « Preferences » et sélectionnez « Android » dans la liste. Dans la case « SDK Location », sélectionnez l'endroit où vous avez installé le SDK Android un peu avant (en général C:\Program Files\Android\android-sdk pour Windows). Validez.

> Android SDK Manager

Il est maintenant temps d'installer les différents éléments Android dont nous avons besoin. Allez dans « Windows » puis « Android SDK Manager ». Vous avez devant vous la liste des packages que Google met à votre disposition. Il n'est pas nécessaire de tous les installer, en fonction de ce que vous voulez faire, mais certains éléments sont indispensables : SDK Tools et SDK Platform-Tools, ainsi qu'au minimum une plateforme (une par version de chaque API Android). Il est très utile d'installer également la documentation et les exemples fournis par Google [Fig.4].

Pour votre choix de la plateforme à installer : choisissez celle sur laquelle vous voulez travailler en fonction du nombre de smartphones sur lesquels vous souhaitez fonctionner. La logique étant que plus vous prenez une API récente, plus elle offre de

possibilités, mais moins votre application sera compatible avec les périphériques anciens. Un exemple concret : si vous voulez utiliser la vue CalendarView pour gérer facilement un calendrier, vous devrez forcément travailler à partir de l'API 11 (indiqué dans la documentation). Donc votre application ne fonctionnera que sur tablettes en 3.X et périphériques en 4.X.

Quelques précisions concernant les différentes versions d'Android : les versions 1.X sont les versions les plus anciennes d'Android. Les versions en 2.X sont des versions pour smartphones sorties après janvier 2010, tandis que les versions en 3.X sont des versions pour tablettes. Cependant, la dernière version, le 4.X, permet de rattacher les deux mondes (des tablettes et des smartphones), simplifiant la vie des développeurs. À noter que vous pouvez également coder pour GoogleTV en 4.0.

Petite précision concernant la version 4.0 : pour pouvoir créer une machine virtuelle d'un périphérique sous Ice Cream Sandwich, vous devrez obligatoirement installer le « ARM EABI v7a System Image ».

Note : si vous choisissez de tout installer, sachez que le volume de données total est assez important (un dossier avec l'ensemble des composants installés peut dépasser les 3 Go), donc l'installation peut prendre du temps, surtout si vous avez une connexion assez lente.

Un dernier point pour terminer : il est important de toujours tenir à jour sa version d'Eclipse, de l'ADT ainsi que des éléments du SDK Android. Pour cela, allez régulièrement (une fois par semaine) dans « Help » puis « Check for updates » dans votre Eclipse.

Pour garder à jour le SDK, lancez l'Android SDK Manager dans le menu « Windows », et

la nouvelle interface vous permettra de voir simplement les nouveaux paquets existants, ainsi que les mises à jour disponibles.

Note : parfois les mises à jour du SDK ne seront visibles qu'après avoir mis à jour votre ADT. Pensez donc bien à les mettre à jour tous les deux (et l'ADT en premier).

> Ressources :

Site officiel de Java :

<http://java.com/fr/download/manual.jsp>

Site officiel de Eclipse :

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

Site de téléchargement du SDK Android :

<http://developer.android.com/sdk>

URL de téléchargement du plugin ADT pour Eclipse :

<https://dl-ssl.google.com/android/eclipse>

Documentation Android officielle :

<http://developer.android.com>

> Conclusion

Vous possédez désormais un environnement de développement Android complet : un IDE (Eclipse), le SDK Android et le plugin ADT qui permet de faire le pont entre votre IDE et le SDK. Il ne vous reste « plus » qu'à créer vos applications. Pour cela, n'oubliez pas les exemples disponibles via l'Android SDK Manager et les explications de la documentation Google (disponible en ligne sur <http://developer.android.com>). Vous pourrez mettre à disposition vos applications sur l'Android Market, ou simplement les conserver pour des utilisations personnelles.

Mickaël Bidon,

Développeur Android

3IE (Institut d'Innovation Informatique de l'EPITA). <http://www.3ie.fr>

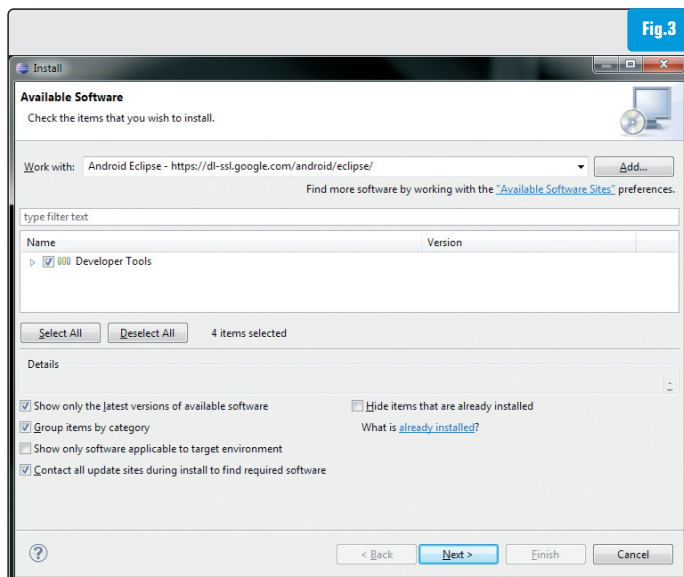


Fig.3

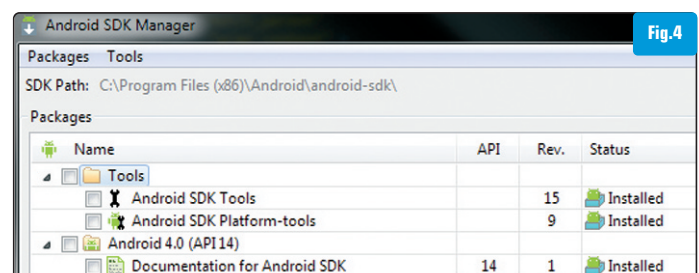


Fig.4

A LIRE DANS LE PROCHAIN NUMÉRO
Février 2012-N°149, parution le 31 janvier 2012

✓ **Windows 8 : Au cœur de WinRT**

✓ **Webmaster :** Traquer les bugs de son site web

✓ **Coding4Fun :** Prendre le contrôle de son PC avec son smartphone

Cahier spécial : Qualité logicielle avec les outils Microsoft

C'EST ENCORE NOËL !

SPÉCIAL
NOUVELLE FORMULE
ET
NOUVELLE ANNÉE

jusqu'à

Abonnez-vous -50%



Exceptionnellement,
recevez sans supplément de prix,
un des livres numériques des Editions Eni,
au choix, sélectionnés par la rédaction
de Programmez,
et d'une valeur de 21€ à 31€
selon les titres !

1 livre Eni* offert

*Version numérique

Html 5 CSS3 (Prix Version numérique : 23,92€)

Google Analytics (Prix version numérique : 21,20€)

Développez une application Android (Prix Version numérique : 31,20€)



Valeur des 11 numéros : 65,50€ + Valeur du livre : 21 à 31€
= Valeur totale : 86€ à 96,70€

Offre : 49€ seulement (ou 59€ avec accès illimité aux archives).

Abonnez-vous à partir de 3,30€ seulement par mois

ABONNEZ-VOUS www.programmez.com Toutes les offres sont en ligne

OUI, je m'abonne

(à retourner, avec votre règlement à :
Groupe GLI, 17 chemin des Boulangers 78926 Yvelines cedex 9

- ☐ Abonnement 1 an au magazine + 1 livre numérique ENI : 49 € (au lieu de 86,45 € à 96,45 € : tarif au numéro 65,45 € + 1 livre)*
- ☐ Abonnement Intégral : 1 an au magazine + archives sur internet et pdf + 1 livre numérique ENI : 59 €*
- ☐ Abonnement 2 ans au magazine + 1 livre numérique ENI : 79 €*
- ☐ Abonnement 2 ans au magazine + archives sur internet et pdf + 1 livre numérique ENI : 89 €*

Livres à Choisir : ☐ Html 5 CSS 3 ☐ Google Analytics ☐ Développez une application Android

Détails sur www.programmez.com/abonnement.php

(*) Tarif France métropolitaine

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

Prénom : _____ Nom : _____

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

(Attention, e-mail indispensable pour le livre numérique et les archives sur internet)

A la découverte de



Windows Azure™

- Ma première application cloud
- Les scénarios d'usage
- SQL Azure et Data Sync
- Windows Azure :
une plateforme ouverte



sommaire

Essayez Windows Azure !.....3

Ma première application
Windows Azure.....4Windows Azure :
une plateforme pour tous
les développeurs.....6Les données avec SQL Azure
et Windows Azure Storage.....8La synchronisation de la donnée
en mode Cloud.....10Windows Azure
pour les sites Web.....12Windows
Azure
et la
mobilité.....14

Windows Azure

PROgrammez !
LE MAGAZINE DU DÉVELOPPEMENT
www.programmez.com

Supplément de PROGRAMMEZ N°148 Janvier 2012 - Ne peut être vendu séparément.

Rédaction : François Tonic, rédacteur en chef. Experts : Florent Santin, Julien Corioland, François Mérand, Jason De Oliveira, Michel Hubert, Safae Rhayour, Jean-Luc Boucho, Jérémie Devillard, Simon Ferquel.**Editeur :** Go-02 sarl, 21 rue de Fécamp 75012 Paris. Directeur de la publication : Jean-Claude Vaudecrane. Dépôt légal : à parution - Commission paritaire : 0712K78366 ISSN : 1627-0908. Imprimeur : S.A. Corelio Nevada Printing, Bruxelles.

édito

Le meilleur du cloud computing

Le cloud computing se décompose principalement en trois couches : le SaaS pour les services applicatifs (Office 365, Online Services), le PaaS pour la plateforme (Windows Azure) et enfin le IaaS pour l'infrastructure (Hyper-V pour le cloud privé, les solutions de virtualisation serveur). Windows Azure est la partie plateforme c'est-à-dire qu'elle cache la complexité de l'infrastructure, son administration, sa mise en œuvre. L'administrateur et le développeur ne s'occupent que du développement, déploiement et production.

Windows Azure repose sur plusieurs services : stockage, moteur de base de données relationnelle, outil de supervision, connecteurs pour créer un cloud hybride, outil d'optimisation de diffusion de contenus (CDN). À cela se rajoutent un kit de développement (SDK), un modèle de développement de bout en bout. Une des forces de Windows Azure est son ouverture vers les applications non .Net : Java, PHP, node.js, Python. Cette plateforme multilingue est un atout précieux pour les développeurs Java qui veulent faire du cloud et s'appuyer sur une offre PaaS reconnue et fiable.

Finalement, à quoi va réellement servir Windows Azure ? De nombreux scénarii sont possibles. Le plus répandu sera le déploiement sur le PaaS d'un site web (e-commerce par exemple). Ce mode de fonctionnement, après quelques adaptations du code et de l'architecture, va permettre au site d'absorber les pics de charge et les périodes creuses ou d'assurer la disponibilité du contenu et des pages web. La force du cloud est de pouvoir réagir à la volée pour mobiliser les ressources nécessaires. C'est l'informatique à la demande. Mais surtout, vous ne mobilisez plus des serveurs inutilement lorsque l'activité est creuse. Vous payez « intelligemment » selon les ressources utilisées. Fini la sous-utilisation de dizaines de serveurs, fini la maintenance de votre salle serveur ! Les applications mobiles utilisent de plus en plus des services cloud tels que le stockage ou les bases de données comme SQL Azure, Azure Storage !

Dans ce cahier spécial, nous allons vous faire découvrir l'univers Windows Azure : SQL Azure, Azure Storage, Data Sync, comment créer et déployer sa première application, utiliser les services Azure pour les sites web et les applications mobiles ! Bref, tout ce que vous devez savoir pour bien démarrer dans le cloud.

Bon voyage dans le nuage.

François Tonic
Rédacteur en chef Programmez !

Essayez Windows Azure !

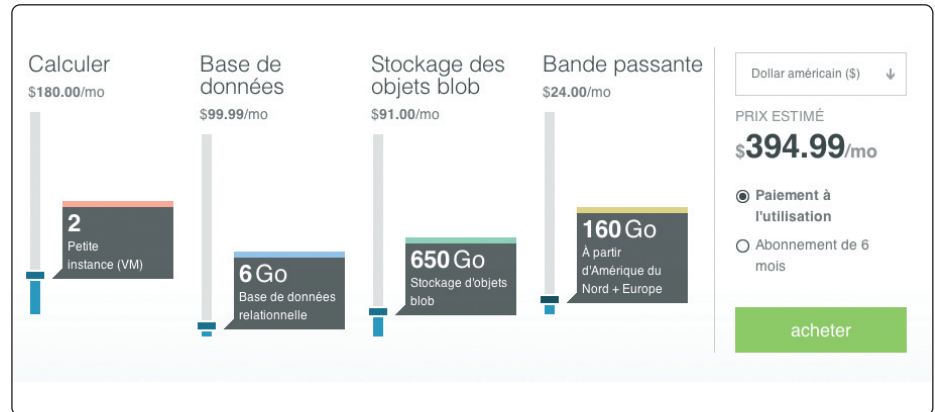


Offre 90 jours gratuits



Téléchargement SDK

Pour tester Windows Azure, vous n'êtes pas obligé de payer. Une offre gratuite de 90 jours est disponible. Pour cela, vous devez disposer d'un compte Live ID, d'une carte bleue valide pour ouvrir votre compte Windows Azure, en quelques clics.



Cette offre spéciale découverte fournit durant 3 mois un quota de ressources Windows Azure capables de répondre à l'ensemble de vos besoins :

- 750 heures / mois de temps de calcul (= instances machines, ressources matérielles en petite instance)
- 20 Go de stockage et 50 000 transactions (chaque mouvement est vu comme une transaction)
- transferts de données : quantité illimitée de données entrantes, 20 Go pour les sortantes
- base SQL Azure 1 Go (web édition)

D'autre part, vous aurez accès aux préversions techniques des futures fonctions comme Data Sync.

Des packages complets !

Comprendre la tarification de Windows Azure nécessite une bonne calculatrice et une connaissance accrue des besoins réels de son application. Vous pouvez payer à la consommation, donc à la demande, ou opter pour des packages tout compris. Microsoft en propose trois :

- Windows Azure : plus orienté plateforme et uniforme dans les ressources disponibles
- SQL Azure : orienté donnée et base SQL Azure
- Stockage : mise avant tout sur le stockage avec 1 To de disponible.

Ces packages sont pour une durée de 6 mois, renouvelable. En cas de dépassement des quotas, vous paierez les ressources supplémentaires. En moyenne, une offre 6 mois permet d'économiser 20 % du tarif à la carte. Cependant, le paiement à l'utilisation peut se justifier

dans certains usages très ponctuels. Les membres MSDN, MPN et BizSpark ont des accès gratuits à Windows Azure pour les aider à adopter la technologie et à tester leurs applications dessus.

Pour vous aidez à déterminer un prix général de votre utilisation de Windows Azure, Microsoft propose une nouvelle calculatrice en ligne. Il vous suffit d'ajuster le type d'instance, la base de données, le stockage en blob et la bande passante pour voir immédiatement l'estimation du coût par mois, en paiement à l'utilisation ou en abonnement 6 mois. Un module avancé permet de calculer l'ensemble des fonctionnalités de la plateforme. Par exemple, un oubli relativement fréquent, pour mettre en place un SLA, il vous faudra deux instances...

Avertissement : avant de vous engager pour un package 6 mois ou à la consommation, vous êtes invité à lire l'ensemble des conditions générales et particulièrement toutes les clauses de qualité de services.

Site français : <https://www.windowsazure.com/fr-fr/pricing/calculator/>

Prendre en compte l'ensemble des éléments de Windows Azure

N'oubliez jamais que Windows Azure est un ensemble de fonctions, de modules qui sont tarifés et le plus souvent soumis à quotas.

Les éléments de base d'un PaaS (ou IaaS) public comme Windows Azure sont :

- le temps de calcul (le fameux compute) : il s'agit des instances sur lesquelles vont s'exécuter vos applications. Il existe différentes

tailles d'instances : très petite, petite, moyenne, grande, très grande. Les différences se feront sur la mémoire, le processeur, le stockage, les performances E/S et le coût horaire.

- Base de données SQL Azure : deux éditions disponibles (web et business). La différence principale se fera sur la taille de la base (à partir de 1 Go)
- transferts de données : en Go, entre régions géographiques, données sortantes (les données entrantes sont maintenant gratuites)
- fonctions de diffusion de contenu (CDN)
- réseau virtuel (Virtual Connect)

Récupérer les SDK

Pour développer sur Windows Azure, il vous faudra télécharger et installer le SDK Windows Azure (actuellement la v1.6). Il fonctionne avec Visual Studio.

Vous pouvez récupérer les SDK pour Java, PHP et node.JS.

Ressources

- Tous les détails de tarification : <https://www.windowsazure.com/fr-fr/pricing/details/>
- Assistance et aide en ligne : <https://www.windowsazure.com/fr-fr/support/contact/>
- Informations générales : <https://www.windowsazure.com/fr-fr/develop/overview/>
- Training Kit : <http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=8396>
- Coach Windows Azure : <http://msdn.microsoft.com/fr-fr/windowsazure/ee621278>

François Tonic

Ma première application Windows Azure

La maturité de Windows Azure n'est plus à démontrer. Il en est de même pour son environnement de développement. Pour vous permettre d'en découvrir la puissance et d'en apprécier sa simplicité, nous allons vous guider au travers des différentes étapes pour créer, puis configurer un projet de type Cloud sous Visual Studio et effectuer un premier déploiement sur Windows Azure.

Avant de commencer, allons faire un petit tour du côté des habituels prérequis. La première étape consiste donc à vérifier et éventuellement installer et configurer votre socle applicatif. Vous aurez besoin :

- Du Framework .Net, en version 4.0, avec le SP1 installé
- D'une version de Visual Studio 2010, avec le SP1 installé
- Du Windows Azure SDK, le kit de développement complémentaire
- Une version de SQL Server 2008 R2 ou SQL Server 2008 R2 Express, afin de simuler SQL Azure si vous envisagez d'utiliser une base de données.

Vous pouvez télécharger le SDK ici : <http://msdn.microsoft.com/fr-fr/windowsazure>, son installation ne pose aucune difficulté. Voici le modèle de développement et de déploiement avec Windows Azure (Fig. 1).

Le SDK 1.6 intègre :

- Des modèles de programmation Windows Azure
- Les API pour lancer les machines virtuelles
- Un environnement de simulation Azure en local
- Les outils de packaging et de déploiement automatique
- Des exemples d'applications et un kit de formation disponible en complément dans le Windows Azure Platform Training Kit (<http://www.microsoft.com/download/en/details.aspx?id=8396>)

Création du Projet sous Visual Studio

La création du projet est très simple : lancez Visual Studio en mode administrateur. C'est important pour pouvoir ultérieurement lancer l'émulateur. Une fois dans Visual Studio, sur la page d'accueil cliquez sur « New Project » ou allez dans « File » → « New » → « Project ». Donnez ensuite un nom à votre projet (Fig. 2). L'étape suivante consiste à ajouter des rôles à votre projet. Windows Azure dispose de machines virtuelles appelées « rôles » pour exécuter les

composants applicatifs. La notion de rôle est nouvelle et propre aux projets de type Windows Azure. Pour tirer parti des bénéfices d'Azure en termes de disponibilité et de montée en charge, une application doit suivre les règles suivantes :

- Une application Windows Azure est construite sur un ou plusieurs rôles
- Chaque rôle doit être déployé au minimum sur deux instances
- Une application Azure doit se comporter correctement si une instance tombe. Il est important de conserver un mode dit « stateless » : les applications Windows Azure ne doivent pas gérer d'état pour fonctionner de façon optimale.

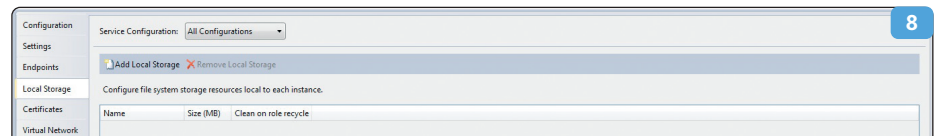
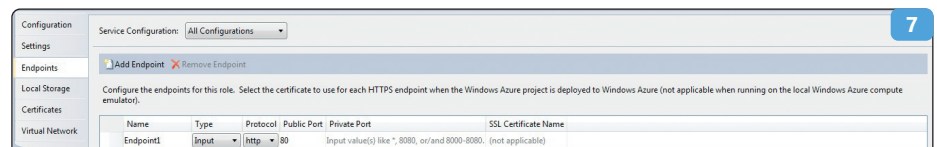
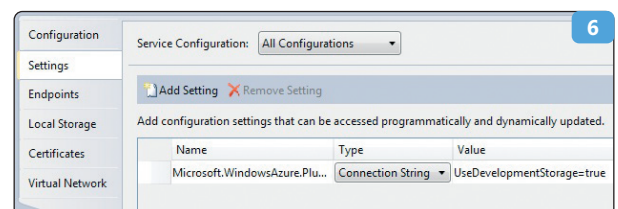
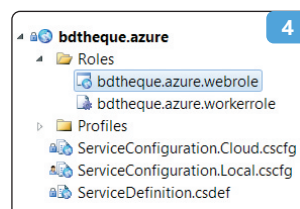
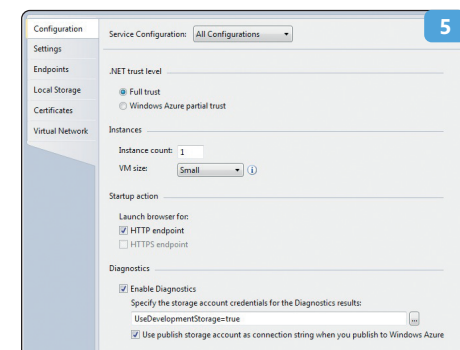
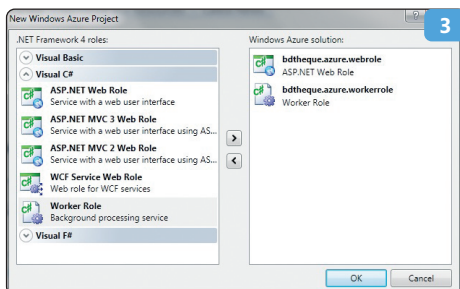
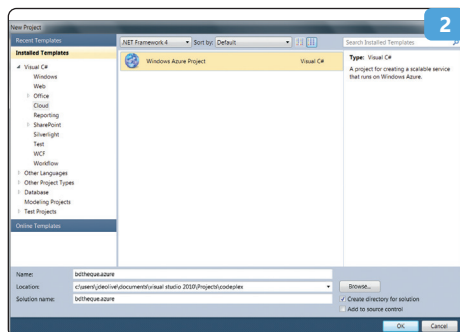
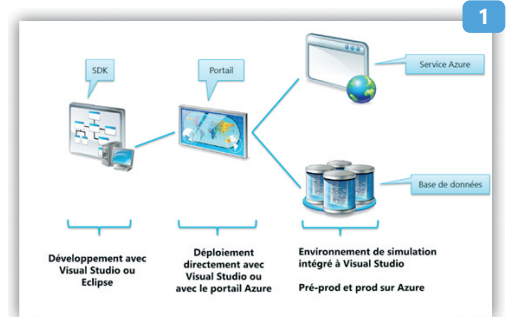
Dans notre exemple, l'application va implémenter deux rôles (Fig. 3) :

- Un Web Role Basé sur IIS, qui exécute les applications Web et utilise ASP.NET, WCF, PHP, etc.
- Un Worker Role utilisé pour les traitements en arrière-plan

Une fois ces deux étapes passées, le projet est créé, et déjà prêt à être déployé. Un projet Windows Azure contient en fait plusieurs projets accessibles depuis l'explorateur de projets :

- Un projet global décrivant les rôles de l'application et leurs configurations.
- Autant de projets que de rôles créés.

Il est tout à fait possible, grâce à l'émulateur



Windows Azure, de lancer notre application en local et la débayer en appuyant sur F5. Grâce à celui-ci il est possible d'exécuter notre projet en local et de simuler les stockages Azure, cela permet donc de tester et débayer le code utilisant celui-ci. Avant de déployer cette première application vide, il est important de s'arrêter sur la configuration de nos différents rôles.

Le Worker Role et le Web Role ont à peu de chose près la même configuration. Pour voir la configuration de notre Web Role, il faut cliquer sur le bouton droit sur l'un de vos deux rôles dans le dossier Roles de notre projet Cloud et sélectionner Properties (Fig. 4).

La configuration se divise en plusieurs parties.

Configuration (Fig. 5) : La première concerne la configuration générale du Role. C'est ici que l'on va définir le nombre d'instances que l'on va utiliser. Ce paramètre est très important, car c'est lui qui va contrôler la notion de « Load Balancing » et de haute disponibilité de votre solution. Il est conseillé de paramétrer plusieurs instances et non une seule. Microsoft ne garantit d'ailleurs un SLA élevé que si deux instances minimum sont mises en production. Un autre paramètre important est la taille de la machine virtuelle, configurée selon la taille de l'application. Les informations relatives aux tailles des machines virtuelles sont disponibles ici : <http://www.microsoft.com/france/windows-azure/offres/tarification.aspx>.

On peut également choisir de lancer l'application en http ou https (si des certificats SSL sont présents sur le portail Azure). Dans le cas d'utilisation d'https, il sera nécessaire d'installer les certificats sur le serveur.

Settings (Fig. 6) : Cette deuxième partie permet de stocker différents paramètres de configuration (chaînes de connexion, emplacement du serveur de stockage Azure, Diagnostics ...).

Endpoints (Fig. 7) : Cette partie permet de définir les points de communication (http, https, ports, ...).

Local Storage (Fig. 8) : Cette partie permet de déclarer sur le File System local de l'instance en production sur Windows Azure des espaces de stockage. (Pour créer un système de cache par exemple).

Certificates : Cette partie permet de rajouter des certificats SSL, qu'il faut avoir installés auparavant sur le serveur via le portail.

Virtual Network : Cette dernière partie permet de gérer l'utilisation d'Azure Connect, qui autorise votre application à se connecter sur un serveur interne à l'entreprise par TCP. Voilà, vous avez maintenant un projet, vide, mais correctement configuré pour pouvoir effectuer votre premier déploiement sur Windows Azure.

DEPLOIEMENT SUR WINDOWS AZURE

Il peut être effectué de deux manières différentes :

- Dans le portail Windows Azure
- Dans Visual Studio 2010 : le processus de déploiement est complètement intégré à l'outil de développement. Il peut aussi être pris en charge dans le processus d'intégration de Team Foundation Server.

Alors prenons le temps de tester les deux options.

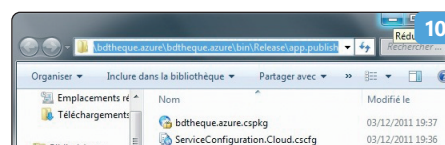
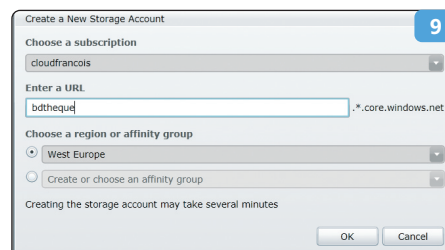
DEPLOIEMENT VIA LE PORTAIL WINDOWS AZURE

Connectez-vous avec votre profil sur le portail Azure. Sur la partie gauche de l'écran sont proposés dans un menu les différents services et actions possible. Pour créer le 1er service, il faut cliquer tout simplement sur l'icône « New Hosted Service ». Qui donne accès à l'écran de configuration. Donnez un nom pour votre service, qui sera donc accessible via l'URL nom du service. cloudapp.net. Pour ce premier déploiement c'est l'option « stage environment » qui est sélectionnée ; il s'agit bien de tester la solution, pas de la mettre en production. La solution sera déployée sur les datacenters Européens de la plateforme Windows Azure.

Ce point étant traité, il faut maintenant s'intéresser à la partie stockage des données : Après avoir cliqué sur « Storage Accounts », puis l'icône « New Storage Account » (Fig. 9).

Une fois cette étape franchie, nous avons donc deux comptes actifs : un pouvant héberger sur plusieurs machines virtuelles nos services applicatifs, et un second pouvant héberger nos machines virtuelles dédiées aux données.

Vous pouvez profiter de cette étape pour créer si vous le souhaitez un serveur SQL Azure et votre première base de données. Les étapes de création et de configuration des ressources Azure sont maintenant terminées, nous allons pouvoir tester la mise en ligne de la solution « vide » : Il suffit d'effectuer un clic droit sur votre projet dans Visual Studio et de sélectionner Package (Fig. 10). Les deux fichiers demandés



par le portail sont générés automatiquement. Il ne reste plus qu'à copier le chemin des fichiers, les coller sur le portail et valider la création du service hébergé. Au bout de 10 à 15 minutes maximum, votre application sera disponible sur le cloud via l'URL de développement ou de production (selon votre choix précédemment effectué) (Fig. 11). Votre première application Windows Azure est en ligne, félicitations !

DEPLOIEMENT VIA VISUAL STUDIO

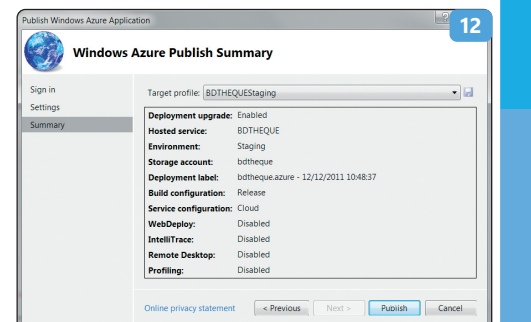
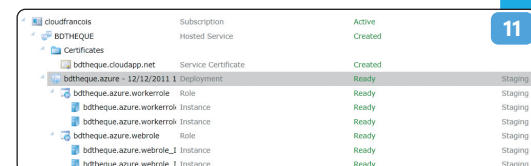
Il existe une deuxième méthode pour déployer une application sur Windows Azure. Via Visual Studio : au lieu de régénérer les packages de l'application et de les uploader sur le portail, quelques manipulations simples vont vous le permettre depuis votre Visual Studio. Pour cela, sélectionnez le projet racine, puis cliquez sur le bouton droit pour sélectionner « Publish ».

Pour le premier déploiement, une configuration initiale doit être effectuée. Il faut en effet que l'on s'authentifie sur le portail via Visual Studio. Ceci se fait via une authentification par certificats, ce qui est bien compréhensible. Ensuite, les écrans de publication s'enchaînent, un premier permettant de sélectionner son abonnement Windows Azure, suivi des paramètres de mise en staging de l'application, avant d'obtenir votre accord final (Fig. 12).

Voilà, vous avez en main les principales étapes pour créer votre première application Windows Azure et la mettre en production. A vous de jouer maintenant !

François Mérand
National Practice Leader – Alliance Microsoft
SOGETI France
<http://www.dotnetrangers.net>

Jason De Oliveira
Practice Manager & Solutions Architect | MVP
C# - SOGETI France
Blog: <http://jasondeoliveira.com>



Une plateforme pour tous les développeurs

La plateforme Windows Azure s'appuie sur des systèmes d'exploitation Windows 2008 et Windows 2008 R2 pour exécuter les services hébergés qui y sont déployés. Le système d'exploitation utilisé est tout ce qu'il y a de plus standard, similaire à la version déployée en entreprise. De ce fait, le serveur web IIS s'y exécute donc nativement et tout code .NET fonctionne naturellement sans problèmes, sans aucune adaptation spécifique.

Mais Windows Server, ce n'est pas que .NET et IIS. Le serveur permet depuis des années d'exécuter beaucoup d'autres serveurs applicatifs pour faire tourner des langages et frameworks non Microsoft. Rien n'empêche donc d'en faire de même avec Windows Azure. Ceci fait même partie des scénarios préconisés, outillés et documentés par Microsoft.

Serveurs applicatifs

La seule contrainte pour faire fonctionner un serveur applicatif non IIS dans Windows Azure est de scripter dans un rôle Azure son installation, sa configuration et son activation. Par exemple, Apache, Tomcat ou Node.JS ont tout trois leurs programmes d'installation et leurs diverses dépendances qui peuvent être exécutées en ligne de commande en environnement Windows. Ceux-ci sont donc tous compatibles avec Windows Azure et peuvent être packagés pour y être exécutés.

Pour faire gagner du temps, certains projets communautaires proposent déjà des modèles de services hébergés dédiés aux serveurs non Microsoft. Ceux-ci mettent à disposition des développeurs un modèle de solution Visual Studio qui contient à chaque fois un projet Cloud chargé de déployer dans un rôle Windows Azure le serveur Web ou applicatif ainsi que le code source à y exécuter. On retrouve notamment sur la plateforme communautaire de partage de code source Microsoft « CodePlex » plusieurs déploiements clé en main pour les célèbres serveurs Apache (par ex : <http://wastarterkit4java.codeplex.com/>) et Tomcat (ex : <http://tomcatazure.codeplex.com/>).

Concernant l'exécution de code PHP ou de Ruby, celle-ci étant nativement supportée par le serveur Microsoft IIS 7 au travers de son module FastCGI, elle ne nécessite donc pas d'installation de serveur complémentaire. Il suffit donc ici de configurer le service hébergé Windows Azure pour packager le code source à déployer et configurer ce module de IIS. Langages et environnement de développement Java, PHP et Ruby peuvent donc être exécutés sur Windows Azure. Mais à quoi bon faire tourner du code sur une plateforme de cloud sans tirer le maximum des avantages de celle-ci ? Dans du code sur Azure il est donc important de pouvoir utiliser, comme en .NET, les différents services proposés par la plateforme. Pour ceci, il est par exemple très intéressant de pouvoir, depuis ces langages, manipuler les données en stockage Windows Azure. En effet, ce mode de stockage persistant propose de manipuler des tables, blobs et files d'attente hautement performantes et disponibles pouvant atteindre des volumétries conséquentes. Celui-ci est accessible

en utilisant le protocole HTTP/REST pour manipuler les données, chose possible depuis n'importe quel langage de développement en implémentant un système de requête HTTP.

Pour simplifier le travail des développeurs qui travaillent sur les langages Web les plus utilisés tels que PHP, Ruby et Java, Microsoft fournit une version de son SDK Windows Azure qui encapsule la complexité d'accès aux données et la restitue sous la forme d'une API très simple et adaptée au mode d'utilisation de chaque langage.

Les deux exemples de code suivants illustrent respectivement en Java et en PHP la manipulation des API de stockage Windows Azure pour télécharger un Blob (binaire) stocké sur le cloud :

```
CloudStorageAccount storageAccount =
    CloudStorageAccount.parse(storageConnectionString);
CloudBlobClient blobClient = storageAccount.createCloudBlobClient();
CloudBlobContainer container = blobClient.getContainerReference
    ("NonContainer");

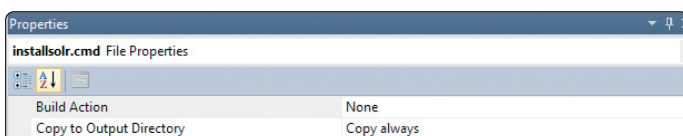
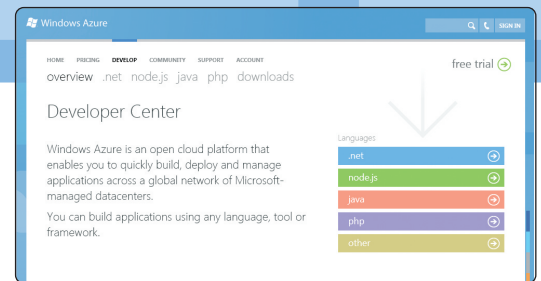
for (ListBlobItem blobItem : container.listBlobs()) {
    if (blobItem instanceof CloudBlob) {
        CloudBlob blob = (CloudBlob) blobItem;
        blob.download(new FileOutputStream(blob.getName()));
    }
}

require 'Microsoft\WindowsAzure\Storage\Blob.php';
$storageClient = new Microsoft_WindowsAzure_Storage_Blob('blob.
core.windows.net',
'NomCompte',
'CleAccess');
$storageClient->getBlob($_POST['containerlist'],
$_POST['bloblist'],
'C:\'. $_POST['bloblist']);
```

L'ensemble des SDK sont référencés et documentés directement sur le site de Microsoft, sur une page dédiée à l'interopérabilité avec Windows Azure : <http://www.microsoft.com/windowsazure/interop/>.

Enfin, les SDK sont certes pratiques, mais le meilleur gain de productivité qu'un développeur peut espérer atteindre se réalise au travers de la qualité de son outillage et donc de son environnement de développement.

Le SDK Azure enrichit Visual Studio en proposant des modèles de projets et des outils pour faciliter le packaging et la publication directement depuis l'environnement de développement. Pour proposer aux développeurs Java et PHP utilisant Eclipse une expérience de développement et de déploiement similaires, les équipes Azure mettent à disposition, toujours au travers de la page Web interopérabilité Azure, un add-in



à Eclipse (l'environnement de développement phare des développeurs Java) disponible en téléchargement gratuit.

Aller plus loin

Windows Azure ne s'arrête pas uniquement aux serveurs et services Web. Dans certains contextes applicatifs, il peut être nécessaire d'héberger d'autres types de serveur : service de cache distribué, base de données NoSQL, serveur de cartographie... Et là, la règle reste toujours la même : à partir du moment où l'installation du serveur peut être scriptée, celui-ci peut être déployé et tourner dans une instance Windows Azure. Techniquement, Windows Azure propose dans la configuration d'un service hébergé d'entamer des tâches de démarrage (que l'on nomme des « StartupTask »). Celles-ci peuvent tourner en tâches de fond, être bloquantes (synchrones) ou non, et sont le point d'entrée idéal pour exécuter un script bat ou powershell. L'exemple de fichier de configuration suivant décrit un exemple de package Azure chargé d'exécuter des scripts à son démarrage :

```
<?xml version="1.0" encoding="utf-8"?>
<ServiceDefinition name="DemoAzure" xmlns="http://schemas.microsoft.com/ServiceHosting/2008/10/ServiceDefinition">
  <WorkerRole name="MonRole" vmsize="Small">
    <Startup>
      <Task commandLine="TelechargePackage.bat" executionContext="elevated" />
      <Task commandLine="ConfigureService.cmd" executionContext="elevated" />
    </Startup>
  </WorkerRole>
</ServiceDefinition>
```

Une fois déployé dans le centre d'hébergement Windows Azure, le package de référence est donc déployé sur autant d'instances de machines virtuelles que nécessaire et, à chaque déploiement et ajout de machine virtuelle, celui-ci va exécuter ses différentes tâches pour installer, initialiser

et configurer le serveur voulu. Enfin, pour les produits dont l'installation peut être plus complexe et ne pouvant pas être installés sous la forme de scripts, il est également possible d'utiliser un rôle de type machine virtuelle pour gérer la compatibilité dans Windows Azure.

Création de Startup Tasks

Avant d'ajouter la startup task dans le fichier CSDEF, il faut créer le cmd associé. Pour ceci, nous devons rajouter ce fichier dans le projet correspondant au rôle. Attention cependant, les fichiers textes créés depuis Visual Studio sont incompatibles avec cmd.exe. Il faut donc les créer avec le bloc-note, puis les ajouter au projet. Une fois ajouté au projet, dans la fenêtre de propriétés du fichier, il faut placer l'attribut « Build Action » à « None », et le « Copy To Output Directory » à « Copy always ».

Dans ce cmd, on pourra alors utiliser des variables d'environnement système comme %~dp0 qui permet de référencer le répertoire du cmd (et exécuter des .exe déployés avec le projet). Par exemple %~dp0MonAppliConsole.exe permet de lancer MonAppliConsole.exe, sous réserve que lui aussi soit en « Copy Always ». On peut aussi utiliser des variables d'environnement définies dans le CSDEF de la manière suivante :

```
<Task commandLine="installsolr.cmd" executionContext="elevated">
  <Environment>
    <Variable name="LOCALSTORESOLR">
      <RoleInstanceValue xpath="/RoleEnvironment/CurrentInstance/LocalResources/LocalResource[@name='SolrStorage']/@path" />
    </Variable>
  </Environment>
</Task>
```

Les valeurs possibles pour l'attribut xpath de RoleInstanceValue, sont disponibles à cette adresse : <http://msdn.microsoft.com/en-us/library/windowsazure/hh404006.aspx>

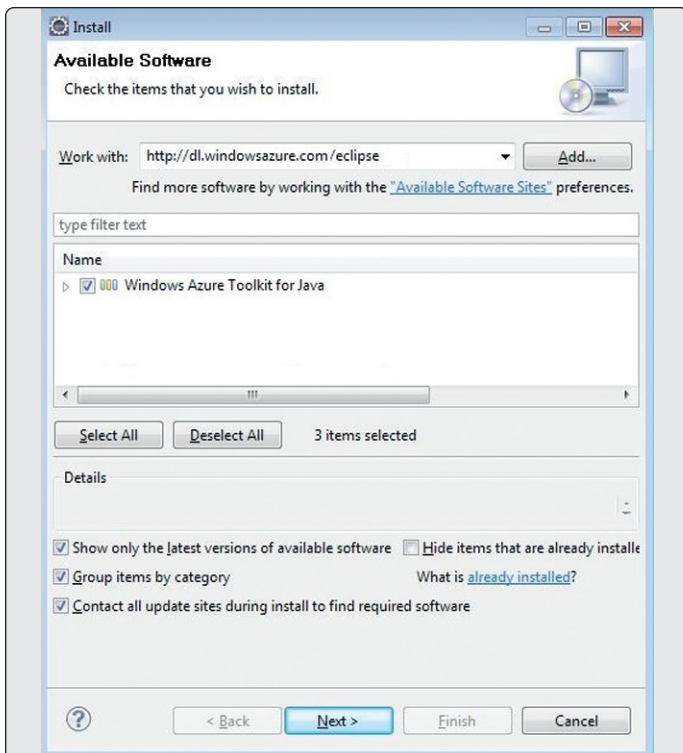
Ressources

node.js : <http://www.windowsazure.com/fr-fr/develop/nodejs/>
 java : <http://www.windowsazure.com/fr-fr/develop/java/>
<http://blogs.msdn.com/b/avkashchauhan/archive/2011/12/18/windows-azure-plugin-for-eclipse-with-java-december-2011-ctp-is-now-available.aspx>
 php : <http://www.windowsazure.com/fr-fr/develop/php/>
 Hadoop : <http://blogs.technet.com/b/dataplatforminsider/archive/2011/12/14/availability-of-community-technology-preview-ctp-of-hadoop-based-service-on-windows-azure.aspx>
<http://www.windowsazure.com/en-us/home/scenarios/big-data/>
 MongoDB : <http://www.mongodb.org/display/DOCS/MongoDB+on+Azure>
 Lucene : <http://code.msdn.microsoft.com/windowsazure/Azure-Library-for-83562538>

Conclusion

Même si les scénarios naturels de Windows Azure sont souvent liés à .NET, cette plateforme est, comme Windows Server, dédiée à tout type d'hébergement. L'objectif de Windows Azure reste de pouvoir faire tourner n'importe quel type d'appliquatif, du plus simple ayant juste un rôle Web aux plus complexes encapsulant une multitude de serveurs et de services. Les récentes annonces de Microsoft sur la prise en charge dans Windows Azure du serveur Apache Hadoop et de Node.JS confirment clairement que les technologies ont une part importante dans le positionnement de la plateforme Azure.

Florent Santin et Simon Ferquel - *Infinite Square*
fsantin@infinitesquare.com - sferquel@infinitesquare.com





Les données avec SQL Azure et Windows Azure Storage

Windows Azure dispose de deux familles de solutions pour le stockage et le traitement des données : SQL Azure et Windows Azure Storage.

SQL Azure est un service de type base de données relationnelles qui possède des capacités très proches de la base de données SQL Server. Accessible par le même protocole (TDS), SQL Azure se développe et s'exploite avec les mêmes frameworks et outils. Vous pouvez donc utiliser : ADO.NET, Entity Framework mais aussi Visual Studio et SQL Server Management Studio pour gérer les bases.

Côté scalabilité, SQL Azure est disponible pour des tailles de 1 à 50 Go actuellement (150 Go pour la fin de l'année), et si cela reste insuffisant, la fédération vous permettra de répartir facilement les données – et l'exécution des requêtes – dans autant de bases que souhaité.

Windows Azure Storage est à l'inverse un service de type NoSQL, nativement réparti et adapté aux très grosses volumétries (jusqu'à 100 To par compte) et proposant trois types de conteneurs :

- Les Blobs : conteneur simple de données brutes, jusqu'à 1 To maximum par blob,
- Les Tables : conteneur d'enregistrements de type dictionnaire (Propriété-Valeur), avec une clé de partition, et jusqu'à 1 Mo par enregistrement,
- Les Queues : file d'attente contenant des messages de 64 Ko maximum.

Ces différents conteneurs sont tous accessibles par une API REST, disponible pour tout client web.

Notez enfin que Windows Azure Storage propose également un disque virtuel appelé Windows Azure Drive : stocké dans un blob, il ne peut être monté en lecture et écriture que par une seule instance de rôle à la fois, ce qui limite son usage à certains scénarios uniquement.

La plateforme Windows Azure propose aussi d'autres services pour vos données, de type Cache :

- **Caching** : service de cache de données, en mémoire et réparti, jusqu'à 4 Go maximum,
- **Content Delivery Network (CDN)** : réseau de distribution de contenu à large bande passant permettant de délivrer les données plus rapidement en étant plus proche de l'utilisateur grâce à un réseau de 24 nœuds répartis

dans le monde.

Ces services sont plutôt à utiliser en complément des services de stockage afin d'optimiser les performances d'accès aux données.

Choisir la bonne solution de stockage

Maintenant que vous connaissez les différentes solutions de stockage, il ne vous reste plus qu'à effectuer le bon choix en fonction des données que vous souhaitez gérer dans Windows Azure. Pour vous aider, voici une proposition de critères à prendre en compte :

- **Nature des données** : s'agit-il de données relationnelles, de fichiers binaires... ?
- **Type de requête** : en lecture et/ou écriture ? Avec un critère simple ou une combinaison ET/OU ? Impliquant une seule entité ou plusieurs (nécessitant une jointure, une transaction...) ? S'appuyant sur des opérations (recherche géo-spatiale, textuelle...) ?
- **Type de client** : le client est-il interactif ? Distant ou dans le même Datacenter Azure ?
- **Type de transfert** : s'agit-il de transférer de gros volumes de données occasionnellement, ou de très petits volumes très souvent ?
- **Sécurité et audit** : les données sont-elles publiques ou doivent-elles être sécurisées ? Avec quelle granularité ? Les accès doivent-ils être audités par la couche de stockage ?
- **Tarification** : SQL Azure revient environ 50 fois plus cher que Windows Azure Storage pour la même quantité stockée, anticiper la volumétrie en production !

D'autres critères peuvent intervenir également, comme des contraintes liées à l'existant, le format de données, les outils tiers ou encore des contraintes réglementaires.

Pour illustrer différents usages et montrer en quoi les différentes solutions sont adaptées, voici quelques scénarios types.

Scénario 1 Migration d'une application Web

Une application web complète a été développée avec les technologies Microsoft : pages web ASP.NET, base de données SQL Server et

certaines fonctionnalités de l'application sont également disponibles via des services web implémentés avec WCF.

L'analyse des données montre que SQL Server est utilisé pour ses fonctions les plus communes, la taille n'excédant pas 10 Go.

Quelle(s) option(s) retenir pour le stockage des données ?

Cette migration devrait tirer parti avantageusement de SQL Azure, de cette manière la couche d'accès aux données sera peu impactée, et les outils d'import/export de données de SQL Server vers SQL Azure tels que SSIS, BCP ou Data Sync faciliteront le basculement des données dans le Cloud.

Quelques remarques tout de même :

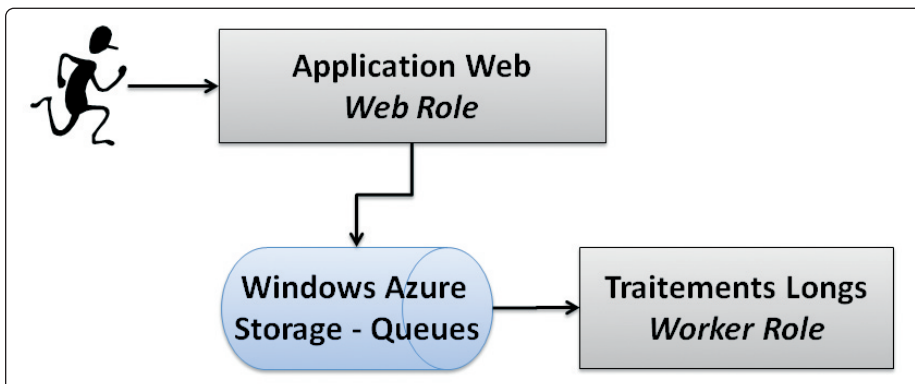
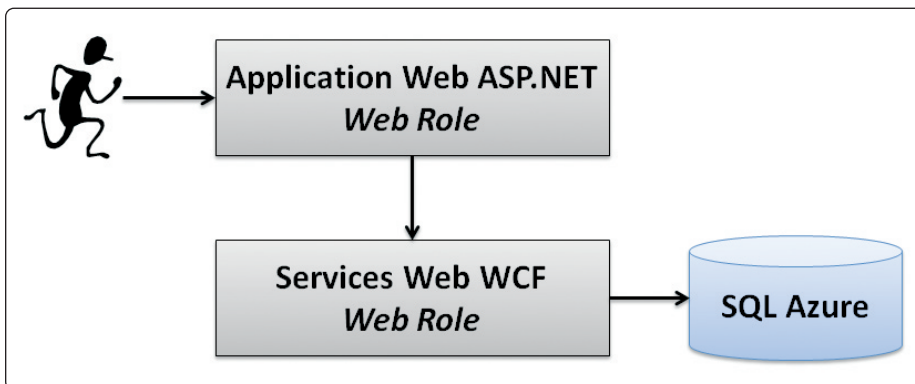
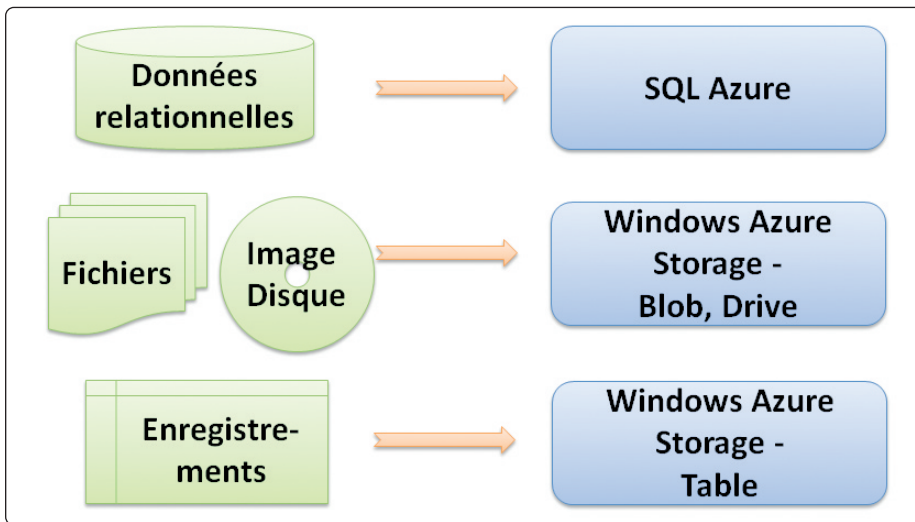
- L'architecture de Windows Azure peut impliquer toute sorte de déconnexions entre les couches applicatives et la base, il est donc préférable de modifier la politique de connexion dans la couche d'accès aux données afin de retenter l'opération en cas d'échec,
- Certaines fonctions de SQL Server ne sont pas disponibles : recherche Full Text, objets CLR, cryptage intégré, données FILESTREAM... vérifier ces éléments avant toute migration, même s'il existe souvent des contournements possibles.

Scénario 2 Traitements longs

Les applications comportent souvent des traitements longs, qui exigent beaucoup de puissance de calcul, ou font appel à de nombreux échanges réseaux.

Voici quelques exemples : traitement d'image (vignettage...), compression de fichiers (création d'archive Zip...), emailing.

Pour éviter de figer l'interface utilisateur, ces traitements doivent être réalisés de manière asynchrone, et il faut donc disposer d'un moyen de stockage intermédiaire – une file d'attente – en attendant le traitement des données en tâche de fond, généralement réalisé par un role Worker.



Quelle(s) option(s) retenir pour le stockage des données ?

En entreprise, utiliser une table SQL en tant que file d'attente est assez fréquent, même si les systèmes Windows incluent Message Queuing (MSMQ) comme solution dédiée. Avec Windows Azure, les alternatives sont semblables, avec les tables de SQL Azure ou de l'Azure Storage, mais aussi les Queues qui sont véritablement adaptées à la situation. En effet, elles implémentent un mécanisme assurant le traitement de chaque message même en cas d'arrêt inopiné de la tâche ou du serveur la traitant, et les messages ont une durée de vie configurable, jusqu'à 7 jours.

Remarque : si la taille du message est supérieure à 64 Ko, une solution de stockage complémentaire (Blobs ou Tables par exemple) sera nécessaire.

Scénario 3 Gestion de documents

Les solutions de partage de documents sur Internet ne manquent pas : Google Docs, Windows Live SkyDrive, Dropbox ou encore Office 365 par exemple. Le scénario actuel consiste à proposer quelques fonctionnalités de gestion documentaire de ce type (upload, partage...) au sein d'une application métier. Côté données, il s'agit de documents de toutes sortes (bureau-

tique, manuels PDF...) et leur description associée (catégorie, rédacteur, date de modification, commentaire, tags...) ; l'ensemble des documents pouvant dépasser 1 ou 2 To.

Quelle(s) option(s) retenir pour le stockage des données ?

Compte tenu des tailles individuelles et cumulées des documents qui peuvent être variables et importants, les Blobs de Windows Azure Storage sont adaptés à cette volumétrie.

Les Blobs possèdent aussi les avantages supplémentaires suivants :

- Les Blobs peuvent bénéficier du CDN qui joue le rôle de Cache à proximité de l'utilisateur
- Les accès aux Blobs peuvent être publics, privés ou partagés de manière sécurisée grâce aux Shared Access Signature (SAS) permettant de définir des droits et durée d'accès précis.

Quant aux informations de description des documents, elles peuvent être stockées soit dans SQL Azure, soit dans les Tables de Windows Azure Storage. Comme nous n'avons pas de détails supplémentaires sur leur usage et leur requêtage, voici les deux options possibles :

- SQL Azure : permet d'apporter une structure relationnelle, fortement typée, et facilitera l'accès aux descriptions et leurs croisements surtout si d'autres données liées résident également dans SQL Azure,
- Tables : convient aussi au stockage des descriptions, tout en apportant beaucoup plus de flexibilité : le nombre de propriétés pouvant varier selon les documents. Si la gestion documentaire est assez indépendante du reste de l'application, les Tables seront une très bonne solution, et assureront le meilleur niveau de scalabilité pour un coût plus faible.

Conclusion

Windows Azure propose plusieurs solutions pour le stockage de données dans le Cloud, avec les deux approches SQL et NoSQL. Choisir la solution optimale requiert une bonne analyse des données et de leur contexte d'utilisation, sécurité y compris. Les scénarios présentés montrent comment s'orienter vers le meilleur choix, voire même combiner plusieurs solutions pour bénéficier de leurs points forts respectifs.



Jean-Luc Boucho
Senior Technology Architect
à Infosys
MCPD Windows Azure

Blog: <http://blog.jeanlucboucho.com>

La synchronisation de la donnée en mode cloud

La donnée est au cœur de tout projet informatique. Sur le web tous les échanges sont sous forme de transactions des données entre « producteurs » et consommateurs de cette même donnée. De fait, l'état de la donnée ne cesse d'évoluer au gré des mises à jour (Add, Update, Delete, ...), ce qui nécessite sa synchronisation entre les différentes structures qui la stockent et la traitent.

La synchronisation de la donnée a toujours été un des contextes d'utilisation du Cloud Computing, ce besoin est d'autant plus important que la plupart des entreprises voient leur Système d'Information se décloisonner et se répartir sur les deux environnements « Cloud » et « On premise » (serveur local). Dès lors, comment garantir que les données soient les mêmes, où qu'elles se trouvent, dans les nuages ou à demeure ?

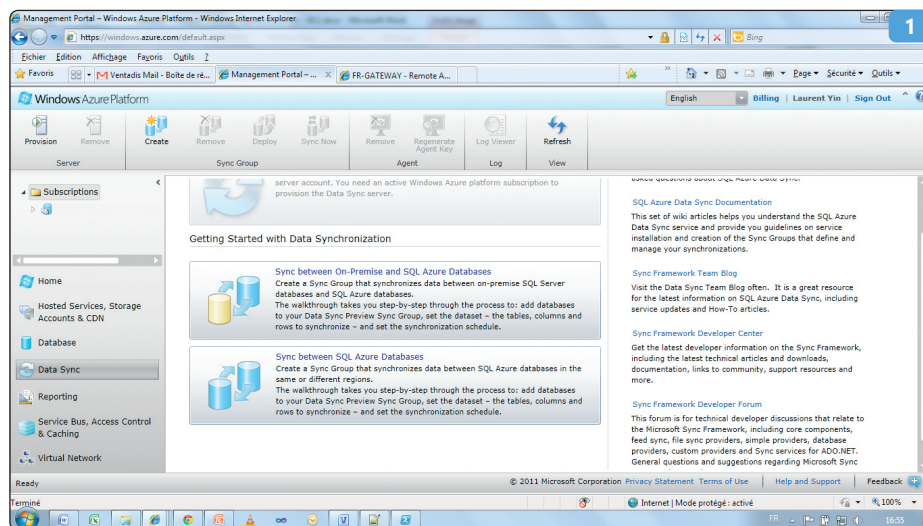
Synchronisation « Cloud / Cloud »

Microsoft propose à travers Windows Azure, un composant de SQL Azure qui permet de synchroniser des données relationnelles uniquement de cloud à cloud ou de cloud à serveur : Data Sync.

Pour mettre en oeuvre une synchronisation, il faut au préalable définir un groupe de synchronisation composé de :

- Une base de données maître dite « Hub Database », obligatoirement hébergée sur un serveur SQL Azure.
- Les bases de données membres du groupe, initialement vides et pouvant être hébergées par un serveur SQL local ou sur Azure.

Lors de la première synchronisation, le service Data Sync va réaliser une copie du schéma de la « Hub Database » vers les bases de données membres. Cela ne concerne néanmoins pas les « Key Constraints », afin de permettre l'insertion des données dans les bases membres dans



n'importe quel ordre. Il est possible de choisir une « synchronisation partielle » en sélectionnant explicitement les tables à synchroniser. Cela permet, entre autres, d'attribuer des permissions pour les bases membres, différentes de celles de la « Hub Database ».

Il est à noter qu'une table de Tracking est créée pour chaque table synchronisée dans un nouveau schéma SQL. Le service « DataSync » rajoute également des « Triggers » et des procédures stockées afin de rassembler et d'appliquer les mises à jour.

Le service offre deux modes de synchronisation :

- Bidirectionnelle : la synchronisation se fait dans les deux sens, ce qui permet d'avoir des bases de données identiques que les modifications soient faites sur l'une ou l'autre des deux bases de données.
- Unidirectionnelle : une base de données de référence est désignée et seules les modifications appliquées à cette base sont répercutées sur la deuxième base.

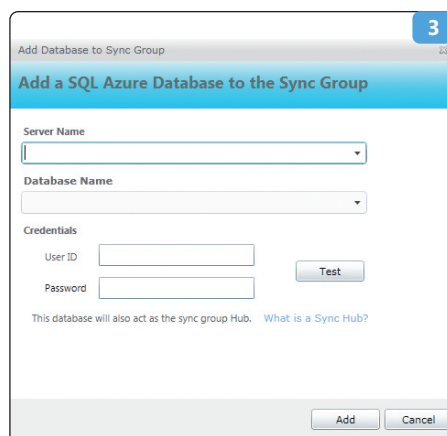
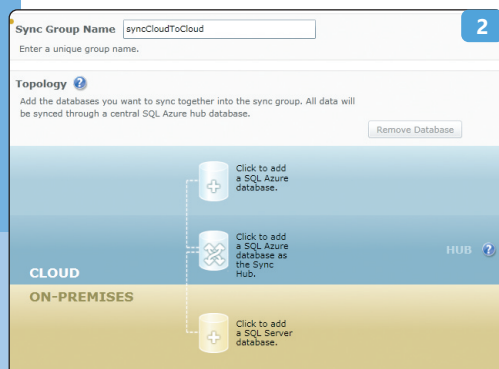
Cas métier

Afin d'illustrer la synchronisation « Cloud To Cloud » du point de vue Métier, prenons l'exemple d'un site e-commerce avec une architecture hybride « Cloud/On premise », car tout n'est pas destiné à aller sur le Cloud ... (fig 7).

Deux scénarios sont envisageables :

Mise en place du PRA

Le PRA, ou le Plan de Reprise d'Activité, permet de maintenir l'activité du site de vente en cas de défaillance d'un des Datacenters de Microsoft. Dans ce scénario, la DSI décide de déployer la solution sur un second Datacenter en Europe en mode « Actif/Passif », en d'autres termes, le site Web hébergé dans le Datacenter « North Europe » ne s'activerait qu'en cas d'anomalie constatée sur celui de « West Europe »



Ouverture à l'International

Dans ce cas de figure, la direction du métier de l'entreprise opte pour une stratégie d'ouverture à l'international en ouvrant un canal de vente en Asie.

Des Web roles supplémentaires sont donc déployés dans le Datacenter de « South East Asia », afin de desservir les internautes de cette région avec des temps de réponse optimaux (grâce au Traffic Manager de Windows Azure). On comprend bien la nécessité de synchroniser les bases Produit entre ces deux Datacenters en respectant les contraintes suivantes :

- Une synchronisation Bidirectionnelle.
- Une synchronisation en Temps Réel.
- Une gestion de conflits optimisée en cas de mises à jour concurrentes.

Configuration de l'outil « Data Sync »

Pour configurer l'outil « Data Sync », il suffit de se rendre sur le portail Windows Azure (<https://windows.azure.com>) et de suivre les étapes suivantes (Fig 1) :

1. Choix du type de synchronisation : Cloud to Cloud.
2. Création du groupe de synchronisation : choix de l'inscription Azure et serveur SQL qui hébergera le groupe de synchronisation.
3. Configuration de la « Hub Database » (Fig 2):
 - a. Choix d'un nom valide du groupe de synchronisation.
 - b. Configuration de la topologie de la synchronisation en ajoutant la « Hub database ».

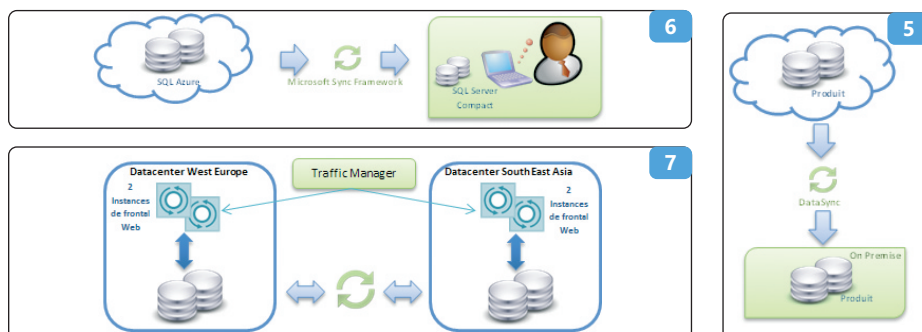
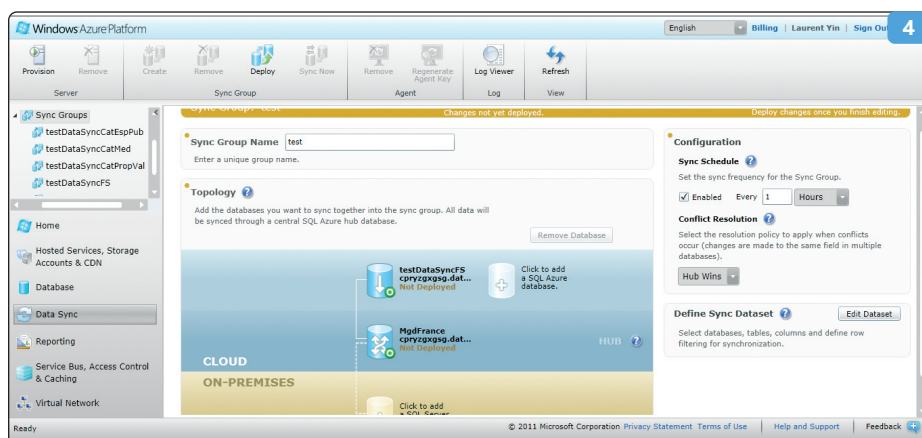
c. Renseigner le nom du serveur SQL Azure, le nom de la base à synchroniser et les paramètres de connexion (Fig 3). A noter que le nom du serveur SQL Azure a le format suivant : <nom_du_serveur>.database.windows.net

4. Configuration de la base membre : on rajoute de la même façon une base de données membre hébergée en configurant le sens de la synchronisation avec le paramètre « set Sync Direction » : mono ou bidirectionnel.
5. Configuration de la synchronisation (Fig 4) : Sur la même page sur le pavé de droite, on peut configurer la fréquence de synchronisation et définir les tables à synchroniser.

Remarque :

- Pour que la synchronisation puisse se faire de manière programmée, ne pas oublier de cocher la case « Enabled ».
- En cas de conflit, c'est-à-dire lorsqu'une donnée est modifiée au même moment dans plusieurs bases de données, il est possible de sélectionner la stratégie de résolution à adopter : soit c'est le hub qui est le maître de la donnée, soit c'est la base membre.
- Lors du choix du set de données à synchroniser, il est possible de sélectionner les tables, les colonnes et les lignes à synchroniser.

Une fois la configuration terminée, il ne reste plus qu'à déployer la synchronisation, il est également possible de consulter les logs grâce au « Log Viewer ».



Synchronisation « Cloud / On premise »

Afin d'illustrer ce scénario, imaginons que le référentiel Produit est géré par une application interne au SI et que les contraintes en termes de temps de réponse sont fortes (Fig 5).

L'application en question doit donc accéder aux données en local, stockées sur une base SQL Server « On premise ». Une synchronisation entre les deux bases « On the Cloud/ On Premise » doit donc être mise en place avec une fréquence donnée.

Les phases de mise en œuvre de cette solution sont pratiquement identiques à celles d'une synchronisation « Cloud To Cloud » mis à part la configuration de la base membre qui, cette fois-ci, est on-premise. Pour ce faire, « Data Sync » installe un agent sur le Serveur SQL « on-premise » afin d'établir une communication via le protocole HTTPS.

Synchronisation en mode déconnecté

Certains utilisateurs, comme des commerciaux, ont besoin d'accéder aux données de l'entreprise en mode déconnecté. Pour ce faire, ils utilisent une application comprenant une base SQL Server Compact.

Dans une situation de mobilité, il n'est pas toujours possible d'avoir accès à Internet pour synchroniser sa base locale de produits avec celle déployée sur SQL Azure.

Dans ce cas précis, on opérerait pour une solution qui permet aux bases de données de se synchroniser dès qu'un réseau internet est à portée, grâce au composant Sync Framework (Fig 6).

Autres outils

La synchronisation entre les environnements « Cloud » et « On premise » peut également être mise en place en utilisant des outils que l'on a l'habitude de manipuler en local.

En particulier, pour les grands volumes de données, on pourrait privilégier des outils de type SSIS (SQL Server Integration Services) ou BCP (Bulk Copy Program) qui prennent nativement en charge des bases SQL Azure.

Michel Hubert,
Manager Logica Business Consulting,
MVP Windows Azure

Safae Rhayour,
Consultante Azure chez Logica Business
Consulting

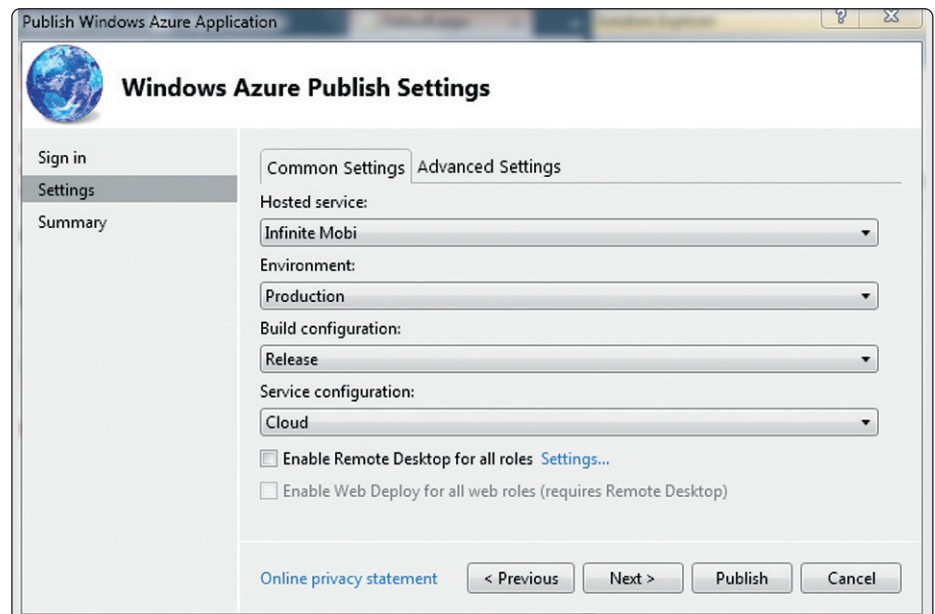
Windows Azure pour les sites Web

De nombreux scénarios peuvent être envisagés sur Windows Azure tels que la mise en place d'agents de calcul distribué, des traitements nocturnes, du stockage volumineux de données, etc. Mais le premier cas d'utilisation qui vient à l'esprit (et sans aucun doute celui qui est le plus souvent mis en œuvre), reste le déploiement et la production de sites web.

Sur le web, il est primordial de disposer d'une plateforme capable de s'adapter à la charge comme le propose Windows Azure. Personne n'est à l'abri d'un effet « buzz » vers son propre site Web, d'une montée brutale de charge. Celui-ci, en créant une charge de trafic considérable, peut rapidement réduire à néant et faire s'effondrer le serveur Web, transformant vis-à-vis de l'utilisateur final un effet positif en image extrêmement négative. Avec Windows Azure, l'administrateur du site Web peut réagir en quelques minutes à ces hausses de trafic en rajoutant des instances de service et ainsi multiplier sa puissance d'absorption de charge à l'infini. Dans certains scénarios avancés, il est également possible de développer des plateformes Web adaptatives ou les instances sont ajoutées / supprimées automatiquement en fonction du besoin à l'instant T.

De plus, au lancement d'un site Web, il est souvent très difficile d'estimer la charge réelle et l'évolution de celle-ci, et donc de dimensionner correctement les serveurs requis. Avec Windows Azure, le risque d'erreur est limité : il suffit de mettre le minimum de serveurs et d'en ajouter en cas de succès.

Enfin, dans le contexte des deux points précédents, Windows Azure est économiquement intéressant. En effet, l'achat de matériel informatique coûte très cher et la location d'un serveur dédié, chez un hébergeur, s'effectue souvent au mois (voire à l'année). Dans le cas d'une infrastructure puissante ayant plusieurs serveurs, le coût de déploiement et d'administration augmente très rapidement (configuration réseau, répartiteur de charge...) et l'ajout de nouvelles machines nécessite un achat de matériel dont il faut absorber le coût et qui peut prendre beaucoup de temps à commander, installer et configurer... De ce fait, il est extrêmement rare de voir un hébergeur supprimer un serveur quand celui-ci n'est plus utilisé ! Dans Windows Azure, le paiement s'effectue à l'heure et à l'instance (au serveur). Si un site Web dispose d'une forte audience uniquement en début de chaque mois, inutile de conserver 4 serveurs tout au long du mois, il suffit d'en conserver 1 seul en période creuse et d'en activer 4 quand cela est nécessaire, en réduisant ainsi le prix de



l'hébergement des périodes creuses par 4... Contrairement à beaucoup d'hébergeurs, sur Windows Azure, le débit bande passante (une taille de tuyaux réservée) n'est pas facturé, mais uniquement le volume de données sortant consommé par mois (nul besoin donc de pré-réserver une bande passante).

Le fonctionnement

Le scénario Web est si classique que les équipes de Microsoft l'ont outillé pour le rendre le plus simple possible. Ainsi, dans un projet de déploiement d'un service hébergé vers Windows Azure, il est possible d'ajouter un rôle Web. Ce rôle a pour objectif de configurer automatiquement Internet Information Services au déploiement de l'instance dans le Cloud Microsoft, en y créant une nouvelle application Web qui pointe vers le code du développeur.

Techniquement, l'appliquatif Web devant être déployé dans Windows Azure doit être packagé dans un service hébergé, qui contient les différents binaires ainsi que deux fichiers de configuration dédiés à Azure.

Ces derniers permettent notamment de déclarer la taille des machines (instances) à utiliser, leur nombre ainsi que la configuration des points d'écoute (ports, protocole, firewall). Au niveau du code ASP.NET en tant que tel, aucune modi-

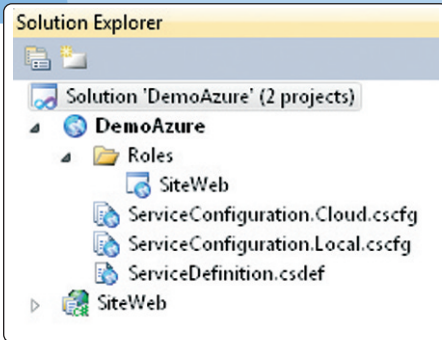
fication n'est à prévoir, un code non Azure peut très bien tourner dans Azure sans aucune modification, juste le packaging compte.

En pratique, certaines modifications peuvent être ajoutées au code telles que l'utilisation du stockage de données Azure, la manipulation des fichiers de configuration Azure ou l'utilisation du service de cache proposé par Azure.

Au niveau du mode de configuration, le fichier de configuration du package Windows Azure permet de définir les points d'écoute à utiliser pour chaque applicatif Web, et de décrire ainsi le port et le protocole de chaque application. L'hébergement d'applicatifs sécurisés via HTTPS ne complexifie pas le déploiement, il suffit juste au préalable de transférer le certificat à utiliser via le portail Windows Azure pour pouvoir l'utiliser dans les applicatifs.

Le scénario est donc très simple : déployer une application ASP.NET dans Windows Azure, avec la configuration par défaut d'Internet Information Services, s'effectue en quelques clics, directement depuis Visual Studio.

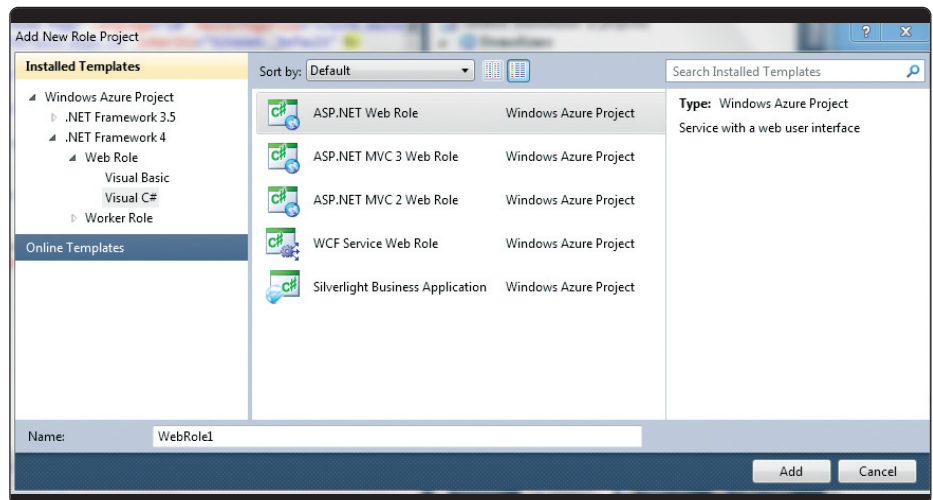
Les scénarios avancés sont également possibles : rien n'empêche un développeur de paramétrer dans son package une configuration avancée de IIS afin d'optimiser son utilisation ou bien d'y faire exécuter du code non Microsoft.



1, 2, 3 : je démarre !

Déployer un site ASP.NET vers Windows Azure s'effectue en quelques clics depuis Visual Studio. La première étape consiste à installer la dernière version du kit de développement Windows Azure (1.6). Celui-ci enrichit Visual Studio de nouveaux modèles de projets, et notamment un de type « Cloud ». Il rajoute également les différents émulateurs permettant de simuler l'exécution de projet et la manipulation de données Azure en local.

Un projet de type Cloud est un projet Visual Studio qui ne contient pas de code, mais uniquement des fichiers de configuration, et qui a comme action de sortie la génération du package à déployer dans Azure. Il fonctionne un peu à la manière d'un projet de packaging MSI. Pour cibler le code à déployer, le projet Cloud déclare des rôles qui ciblent les différents projets de la solution. Chaque rôle représente une instance de serveur différent. Un projet Cloud peut donc à la fois déployer ensemble plusieurs serveurs, par exemple pour isoler sur des machines virtuelles différentes et à des adresses différentes un frontal Web et son site d'administration. En configuration avancée, il est également possible de déployer plusieurs sites Web en mode mutualisé sur une même instance Windows Azure, en jouant sur les noms d'hôte HTTP



pour isoler les différentes applications. Pour un projet Web existant à déployer dans Azure, il suffit donc d'ajouter dans la solution Visual Studio un projet de type Cloud, puis de le configurer pour cibler l'appliquatif Web. Pour ceci, il suffit de déclarer dans le projet Cloud un nouveau rôle, de type Web, et d'y rattacher l'appliquatif à packager. A l'exécution du projet Cloud, le package Azure sera automatiquement généré et déployé sur la fabrique locale Windows Azure – serveur Web d'émulation de l'exécution en mode Cloud. Le même package peut ensuite être déployé dans le data center de Microsoft, directement depuis Visual Studio ou via le portail d'administration de Windows Azure.

Quelques pièges à éviter

Bien qu'un simple site ASP.NET puisse tourner très rapidement dans Windows Azure, certaines bonnes pratiques sont à respecter pour parfaire l'utilisation de cet environnement. Windows Azure est un mode d'hébergement scalable, c'est-à-dire qu'une instance peut être amenée à

tout moment à se dupliquer. Du coup, les applicatifs Web doivent être indépendants de leur machine d'exécution et ne pas utiliser le système de fichier local pour sauvegarder de la donnée. Un système de stockage persistant, tel que le stockage Windows Azure sera plus adapté. Le système de fichier peut toutefois être utilisé pour du stockage temporaire non garanti. De même, les applications Web tournant sur Azure ne doivent pas être dépendantes de la session HTTP, à moins que la gestion de celle-ci ne soit centralisée dans un serveur ou système de stockage partagé.

Scénarios avancés

En mode avancé, la plateforme Windows Azure offre des services complémentaires intéressants : un système de cache distribué, sur lequel un nombre illimité d'instances peut s'appuyer pour augmenter le temps d'accès à certains types de données. On peut également mettre en place des scénarios de fédération d'identités afin de permettre en très peu de lignes de code, d'intégrer dans n'importe quel site Web un mode d'authentification via des fournisseurs d'identités tiers tels que Facebook, Yahoo, LiveID ou Google.

En conclusion

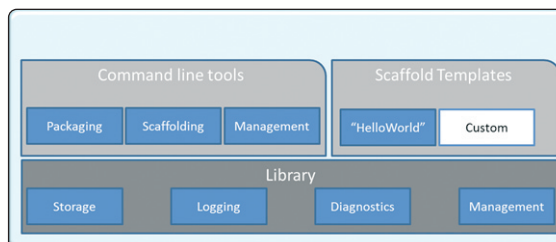
Windows Azure est parfaitement adapté pour les scénarios d'hébergement Web, car le Cloud Microsoft a été pensé pour adresser ceux-ci en premier lieu. Les scénarios classiques tels que l'hébergement de site ASP.NET sont très simples dans Azure, et les scénarios complexes tels que l'exécution de PHP ou la configuration avancée de IIS restent adressables par configuration.

Florent Santin et Julien Corioland
Infinite Square
fsantin@infinitesquare.com
jcorioland@infinitesquare.com

Site PHP sur Windows Azure !

Vous développez des sites dynamiques en PHP ? Pas de problème, vous pouvez coder et déployer vos sites sur Windows Azure. Pour ce faire, vous devez installer Windows Azure SDK for PHP qui permet d'utiliser PHP sur le PaaS de Microsoft. Ce SDK est open source et fournit des outils en ligne de commande, différentes bibliothèques, des templates, des exemples et la documentation technique. F.T. Pour en savoir plus :

<http://azurephp.interoperabilitybridges.com/articles/setup-the-windows-azure-sdk-for-php>
<http://blogs.msdn.com/b/benjamin/archive/2011/12/08/install-the-windows-azure-sdk-for-php-with-windows-azure-sdk-1-6-install-le-sdk-windows-azure-pour-php-avec-le-sdk-windows-azure-1-6.aspx>



Windows Azure et la mobilité

Notre quotidien est un monde d'appareils connectés et interconnectés : Ordinateurs, notebooks, tablettes, Smartphones et maintenant Smart TV, l'ensemble des périphériques qui nous entourent sont liés. L'explosion des applications pour les périphériques portables a permis un essor de cette technologie, notamment, grâce à un grand nombre d'outils, de frameworks et de ressources mis à disposition des développeurs.

Dans l'univers Microsoft, nous allons voir comment Windows Azure peut nous permettre de relever les défis de la mobilité sur les aspects suivants : Web, Services et Données. La première brique à prendre en compte dans l'utilisation de la mobilité est la brique Web : l'accès au site. Les sites internet sont facilement accessibles sur les plateformes mobiles sans ajout de nouvelles applications, l'utilisation du navigateur suffit. Cependant, de par ses dimensions, le rendu de l'information ainsi que l'expérience utilisateur ne peuvent être les mêmes sur un ordinateur et sur un appareil mobile (Smartphone ou Tablette). Concernant la mobilité et le web, la plateforme Windows Azure de Microsoft nous permet d'avoir accès à l'ensemble de la technologie .Net aux travers des Web Roles pour héberger nos applications web. Il est donc nécessaire de remanier son site pour permettre un accès « multi-device », le « web-responsive » est obligatoire pour assurer la meilleure expérience utilisateur, peu importe l'appareil utilisé. Seule la partie frontale et graphique pourra être remaniée. L'ensemble des éléments constituant le back end n'étant pas impacté. Les différentes vues sont en générale basées sur les tailles d'écrans ainsi que sur les caractéristiques des navigateurs. Les étapes du processus permettant l'affichage adéquat du site sont en général les suivantes :

- Détection de l'appareil faisant la requête : type de navigateur, résolution, type d'appareil mobile
- Choix du rendu et redirection vers la vue correspondant à l'appareil ou au navigateur.

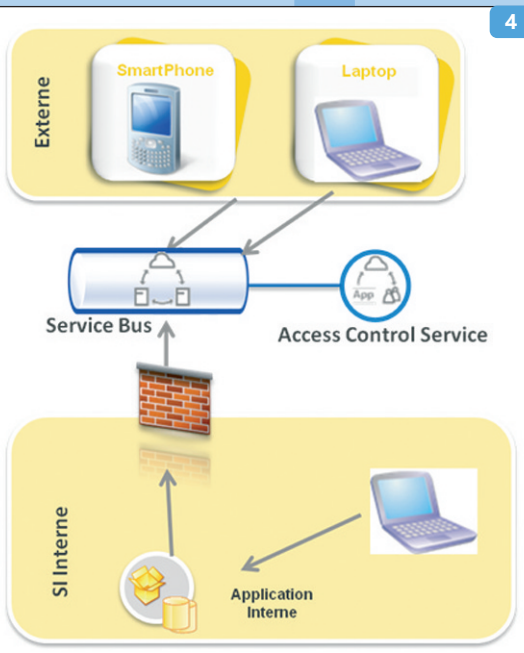
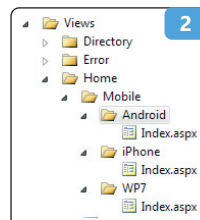
La technologie ASP.Net MVC 3 nous fournit « View Engine » afin de nous permettre de gérer les différentes vues disponibles sur notre site internet en fonction du périphérique. Après avoir installé le package nugget MobileViewEngines, il suffit de définir les différents moteurs que l'on souhaite prendre en compte (Fig 1). Ensuite on réalise l'arborescence adéquate et on crée les pages de rendu pour chacun des devices (Fig2). Cela nous permet d'organiser notre solution web en séparant chaque interface graphique par rapport aux spécificités de son périphérique. Cela oblige à faire des interfaces graphiques propres à chaque périphérique,

```
protected void Application_Start()
{
    AreaRegistration.RegisterAllAreas();

    RegisterRoutes(RouteTable.Routes);

    ViewEngines.Engines.Clear();
    ViewEngines.Engines.AddMobile<WebFormViewEngine>("iPhone", "Mobile/iPhone");
    ViewEngines.Engines.AddMobile<WebFormViewEngine>("Android", "Mobile/Android");
    ViewEngines.Engines.AddMobile<WebFormViewEngine>("WindowsPhone7", "Mobile/WP7");
}
```

1



4

ce qui peut être un travail long et périlleux si l'on ne possède pas les connaissances sur l'ensemble des périphériques mobiles disponibles.

Pour cela, des frameworks nommés Mobile Web Framework sont disponibles pour aider les développeurs, les plus connus sont jQueryMobile (<http://jquerymobile.com>), iUI (<http://iui-js.org>), iQTouch (<http://jqtouch.com>).

La mobilité dans la communication avec les services

Dans le domaine du service, il est important de ne pas chercher à redévelopper un service existant pour lui permettre d'être accessible par l'ensemble des périphériques connectés existants. Les technologies concernant les services Web sont connues et ont évolué pour devenir, tout d'abord, de plus en plus descriptives - avec le SOAP - avant de redevenir plus concises avec la technologie REST puis Json afin de d'optimiser les temps de réponses avec des messages plus concis, plus adaptés aux bandes passantes des périphériques mobiles. Les technologies ASP.Net et WCF nous permettent, côté serveur, d'exposer les services en Soap, en REST et en Json en utilisant les serializers/déserializers adéquats.

iPhone / iOS

Dans le monde Apple, des frameworks sont disponibles :

- Json framework permet d'effectuer mes requêtes en objective C à partir de mon Iphone vers mes méthodes exposées en REST (<http://code.google.com/p/json-framework/>)
- wsdl2objc permet quant à lui de consommer un service SOAP (<http://code.google.com/p/wsdl2objc/>)

Android

Dans le monde Android, les deux principaux frameworks sont :

- google-gson pour les services REST (<http://code.google.com/p/google-gson/>)
- ksoap2-android pour les services SOAP (<http://code.google.com/p/ksoap2-android/>)

Windows Phone 7.x

Concernant les applications sur Windows Phone, des frameworks aussi sont disponibles :

- restSharp (<http://restsharp.org>)

Les applications mobiles ont donc la possibilité de communiquer avec des services Web en REST (et Soap), mais que m'apporte Windows Azure dans ce type de scénarios. L'élément important à prendre en compte ici est que l'ensemble des API Windows Azure est disponible sous forme de

services REST. Et il est tout à fait possible (et aisé) de profiter des services de stockages Azure à l'aide de points de terminaison REST. L'exemple suivant montre une requête GET pour lister les containers du compte de stockage « mystorageaccount » (Fig3). Le résultat est alors fourni sous forme XML. Afin également de faciliter le développement, Microsoft a mis à disposition un Toolkit de développement Azure pour les plateformes mobiles iOS et Android. Les Toolkits sont disponibles aux adresses suivantes :

- <http://github.com/microsoft-dpe/wa-toolkit-ios>
- <https://github.com/microsoft-dpe/wa-toolkit-android>

Ces toolkits simplifient l'accès aux composants Windows Azure tels que les Blobs ou les tables.

Et l'entreprise

Dans les environnements d'entreprise, les technologies énoncées précédemment rassurent car elles démontrent que les web services SOAP seront réutilisables. Mais mon Service Web est à l'intérieur de mon Système d'Information... Comment le rendre disponible hors des frontières de mon S.I. pour les applications mobiles de mes employés et/ou de mes clients et partenaires ? Windows Azure nous fournit dans ce cas le Service Bus, composant de Windows Azure AppFabric, qui va nous permettre de faire la jonction entre mon service Interne et l'ensemble de mes périphériques de manière sécurisée (SmartPhone ou Laptop par exemple). Nous pouvons exposer nos services existants en utilisant le mode « Relay » du service Bus. Access Control Service nous permettra de gérer l'identification des appelants et les autorisations associées (Fig 4).

La mobilité dans l'accès à la donnée

Nous avons vu précédemment comment accéder à nos services web à partir des périphériques mobiles. Comment faire pour donner accès à mes utilisateurs à des données sans redévelopper de nouveaux services d'accès ou modifier mes modèles de données. Dans Windows Azure, les unités de stockages de l'information sont les suivantes :

- Azure storage avec les blobs, les tables
- SQL Azure pour les bases de données relationnelles

Le premier point a déjà été évoqué précédemment et les frameworks Rest nous permettent d'ores et déjà d'accéder à ces ressources.

Le protocole OData

OData (Open Data Protocol) est un des choix disponible pour accéder à différentes ressources de données. OData est en effet un protocole Web pour effectuer des requêtes et mettre à jour des données en débloquent la donnée des silos applicatifs pouvant exister actuellement dans les entreprises. Basé sur des standards Web tel que http, le protocole AtomPub (Atom Publisher Protocol) et JSON, il permet de rendre disponible l'information à partir d'une grande variété d'applications, services ou stockages (Base de données). Quelques exemples d'applications utilisant Odata : SharePoint Server 2010, SQL Server Reporting Service, PowerPivot ...

Conclusion

Avec le développement des terminaux mobiles et leur adoption progressive dans le monde professionnel, le nomadisme en entreprise devient un enjeu majeur, ouvrant des perspectives nouvelles en termes d'organisation du travail et de nouveaux usages. ! Windows Azure apporte un certain nombre de réponses, à travers différents scénarios, pour exposer la donnée, pièce

Windows Azure Toolkit : vive la mobilité !

Vous l'aurez compris, Windows Azure sert aussi aux applications mobiles. Ce n'est pas un hasard si Microsoft travaille activement à fournir un toolkit (Windows Azure Toolkit), pour les principales plateformes mobiles du marché : Android, iOS, Windows Phone.

Aujourd'hui, les applications mobiles sont de plus en plus complexes, nécessitant toujours plus de ressources (pour les traitements, le stockage). Windows Azure peut intervenir comme back-end à votre application mobile, par exemple, pour héberger une partie de l'application, le stockage de données, la base de données. Là, vous évitez de devoir gérer ces services sur le téléphone surtout si votre application fonctionne sur différents terminaux.

Avec les Windows Azure Toolkit for iOS, for Android et for Windows Phone, Microsoft fournit les services, les API nécessaires pour mettre en plus les connexions entre l'application et les services Azure. Et les toolkits évoluent régulièrement. Ainsi, dans la v1 pour iOS, le développeur pouvait uniquement utiliser le stockage dans les tables ou blobs de Windows Azure Storage. Dans la version 1.2.x, le développeur dispose d'un outil de configuration, d'un support pour l'ACS (Access Control Service). Un des challenges fut de simplifier le déploiement des packages sur Windows Azure pour que l'application puisse fonctionner. Le toolkit dispose d'un mécanisme très pratique : cloud ready packages for Devices. Car chaque toolkit mobile respecte le modèle de développement du smartphone : Xcode et Objective-C pour iOS, Eclipse – Java pour Android, Visual Studio pour Windows Phone. D'ailleurs, le toolkit sur Windows Phone se présente maintenant aussi sous la forme de packages NuGet qui permettent d'ajouter des fonctionnalités indépendantes les unes des autres (push, authentification, stockage, ...) à une application Windows Phone existante.

Grosso-modo, les services proposés par tous les toolkits (avec des adaptations pour chaque plate-forme) : authentification, notification, notification de push, SQL Azure, Azure Storage

Ressources

Portail général : <https://github.com/microsoft-dpe>

Toolkit pour iOS : <https://github.com/microsoft-dpe/wa-toolkit-ios>

Toolkit pour Android : <https://github.com/microsoft-dpe/wa-toolkit-android>

Toolkit Windows Phone : <http://watwp.codeplex.com/>

François Tonic

maîtresse de nos systèmes d'information, dans un contexte d'interopérabilité. La transformation du système d'information s'explique également par le besoin de plus en plus fort de faire communiquer l'informatique métier avec des plateformes web telles qu'un portail Intranet, un site d'e-commerce, un extranet proposant des services en ligne aux partenaires.

Michel Hubert,
Manager Logica Business Consulting, MVP Windows Azure

Jérémy Devillard,
Consultant Azure chez Logica Business Consulting.

```
GET http://mystorageaccount.blob.core.windows.net/?comp=list HTTP/1.1
```

3

```
x-ms-version: 2011-08-18
x-ms-date: Sun, 25 Sep 2011 22:08:44 GMT
Authorization: SharedKey mystorageaccount:Eby8vdM02xN0cqFlqUwJPLlmEtlC
```

Concentrez-vous sur vos applications,
nous nous occupons de l'infrastructure.

- **Hébergez** et **exécutez**
vos applications dans le Cloud.
- **6 centres de données** dans
le monde (2 en Europe).
- **99,95 %** de disponibilité.
- Paiement à l'usage.

Essayez gratuitement Windows Azure pendant 90 jours sur
www.windowsazure.fr



Pour utiliser ce flashcode,
téléchargez l'application
gratuite sur <http://gettag.mobi>

Rejoignez-nous sur :



@WindowsAzureFR



<http://www.facebook.com/windowsazurefrance>

Profils d'étudiants Open Source

Si l'Open Source est à la fois au coeur de la formation des développeurs, mais dans le même temps le parent pauvre des organismes de formation, elle reste un objet passionnel.



PHP, RAILS, Linux... quel développeur ou étudiant ignore aujourd'hui ces outils, librement accessibles, et qui bénéficient à la fois d'une riche documentation et d'une commu-

nauté active à même de retenir son attention et de l'accompagner éventuellement dans ses projets. Le Libre et l'Open Source sont omniprésents dans la boîte à outil du développeur comme de l'administrateur. Mais qu'en est-il pour sa formation ? Distinguons tout d'abord deux profils de candidats à la formation : l'étudiant, qui suit un cursus de type scolaire, et le stagiaire, développeur ou administrateur professionnel. Le premier va s'adresser aux écoles d'informatique, tandis que le second se tournera vers les organismes de formation. Dans chacun des cas, l'approche de l'Open Source, la pédagogie et le catalogue des formations seront différents.

> L'étudiant passionné - l'Open Source ADN

Il est rare que l'étudiant de n'importe quel domaine de l'informatique n'ait pas touché une ligne de code. Il suit un cursus de formation basé sur un socle généraliste qui lui fera approcher divers environnements et outils, dont ceux de l'Open Source et du Libre. Il devra également très certainement produire au moins un projet au cours de son cursus. L'Open Source, accessible et documenté, s'impose ici. Comme le constate Julien Ballet, Responsable du laboratoire Logiciel Libre d'Epitech, « 75% de nos étudiants utilisent de l'Open Source, beaucoup PHP ou RAILS, et avec la communauté. »

Autre constat, pour beaucoup d'étudiants qui intègrent les écoles d'informatique, le rapport à l'Open Source est passionnel. Le développement est un vrai passe-temps, en plus d'un métier en devenir. L'école est également un espace de créativité, loin des

contraintes industrielles qui n'offrent pas toujours l'opportunité et le temps de développer un projet comme on le souhaite et avec une recherche de qualité. L'étudiant développeur devra trouver dans l'école un espace qui répond à son projet et qui lui permet de partager expériences et compétences. Au delà du programme imposé, la mission de l'école est ici de sensibiliser aux technologies et aux outils, dont ceux de l'Open Source, ainsi qu'aux méthodes de développement et de travail. « Nous devons sensibiliser aux pratiques usuelles dans le monde de l'Open Source, et à la communication », souligne Julien Ballet, dont le Lab-Free accueille les étudiants à leur demande, pour agir au cas par cas plutôt sur leurs projets de fin d'année ou de fin d'étude, ou sur les projets de R&D dans le cadre de pôles de compétitivité. « Nous sommes à la disposition des étudiants qui veulent s'intégrer dans le monde de l'Open Source afin de contrecarrer certaines mauvaises habitudes... »

> Le stagiaire professionnel - l'Open Source outil

A l'auscultation des catalogues des organismes de formation, force est de constater que l'Open Source fait figure de parent pauvre. Ou alors il se cache... C'est le cas par exemple des formations d'administrateur, un domaine que tout développeur doit au moins approcher, où le Libre est noyé dans une masse d'outils plus ou moins

intégrateurs des solutions qui y trouvent une source de revenus.

> Développer en open, un modèle économique

Pourquoi former les développeurs à l'Open Source ? Interrogeons-nous sur la motivation des éditeurs à intégrer des outils libres et à recruter des développeurs disposant de compétences sur l'Open Source. Depuis le début du siècle, donc après des décennies de balbutiements communautaires, l'approche Libre a commencé à sérieusement séduire les éditeurs comme les entreprises par la pertinence de son modèle économique. Ils se sont découvert de nombreux intérêts à développer en open : la gratuité de la licence en est un, tout comme la présence de riches communautés qui offrent un accès à du développement au coût réduit et qui leur permettent de développer des services. L'utilisation de logiciels libres permet également de répandre rapidement des technologies. « Nous assistons à une transition vers une industrie de services, avec des modèles moins fermés et qui mettent en avant la pertinence économique du modèle de l'Open Source », commente Julien Ballet : « La valeur ajoutée n'est plus le logiciel mais le service qui l'accompagne ». Certains outils Open Source sont par ailleurs « poussés » par les entreprises. Ces outils sont généralement des projets matures, car les entreprises sont rarement prêtes à intégrer

au sein du SI des outils qui n'ont pas atteint une masse critique d'utilisateurs et ne disposent pas d'un écosystème pour assurer le support, la maintenance et la formation. La compétence sur des outils de développement et

“ 5 à 10 % des étudiants d'Epitech franchissent le pas : ils contribuent à des projets et publient du code. ”

matures voire banalisés. Il est vrai que cela correspond à une réalité sur le terrain, où les composants Open Source sont désormais si présents que l'on en oublierait presque leur origine. Par ailleurs, la formation fait partie du modèle économique de l'Open Source... Elle est donc principalement assurée par les distributeurs et/ou

des solutions Open Source peut donc se révéler être un atout supplémentaire sur un CV. Face à une relative uniformisation des formations initiales, et de leur inadéquation cyclique avec les attentes du marché, cet apport supplémentaire peut faire la différence. Une fois passé le cliché du développeur dans son garage, qui a marqué deux géné-

rations de technophiles et n'a pas fini de faire parler de lui, se former à l'Open Source démontre aujourd'hui la capacité du développeur à lancer un projet et à le faire aboutir, ce qui représente une réelle valeur ajoutée.

> Travailler les spécificités du monde Open

Qu'il s'agisse du développeur qui se forme, de l'école ou de l'organisme de formation qui l'accueille, ou encore de l'entreprise qui le recrute, une attention particulière doit être portée aux bonnes pratiques liées à l'Open Source. En la matière, les organismes de formation ont trop peu de contacts avec les entreprises et manquent de réalisme sur les méthodologies de formation. Ces dernières doivent en particulier inciter à faire preuve de plus de transparence et à communiquer sur le projet et son développement. Le développeur doit pouvoir anticiper, expliquer ce qu'il fait et pourquoi il le fait. « *Ce n'est pas inné !* », constate Julien Ballet. « *Une vraie transparence des processus de développement et de la communication devrait être faite. De même, la publication du code et sa mise à la disposition de n'importe qui ajoute une pression pour bien faire les choses et se dépasser.* » Au delà du développement et de l'usage des outils Open Source, si l'apprentissage mène à la création et à la fourniture du code, un vrai défaut demeure dans les pratiques des développeurs ou étudiants qui intègrent les écoles : la capacité à lire du

Conseils aux candidats en formation

Julien Ballet, Responsable du laboratoire Logiciel Libre d'Epitech, donne quatre conseils aux développeurs, étudiants et stagiaires en formation sur les produits Open Source :

1 - Répondre à un besoin propre

Pour qu'un projet Open Source lancé par un étudiant soit un succès, il faut d'abord qu'il le développe pour lui, avec une vraie motivation et en réponse à un vrai besoin. Cette motivation doit être suffisante pour mener son projet à terme.

2 - Ne pas chercher à réinventer la roue

Un des principes de l'Open Source est la ré-utilisabilité des composants logiciels. Mais la mutualisation des efforts n'est pas toujours gratifiante, tout comme la documentation, le test, l'assurance qualité, etc. L'étudiant a tendance à tout refaire, repartir à zéro. Nous assistons à un phénomène de fragmentation, avec de plus en plus d'outils qui font la même chose. Le développeur doit au contraire trouver des solutions qui existent et les réutiliser, et regarder dans le code les choses approuvées.

3 - Communiquer

Beaucoup de développeurs aiment ce qu'ils font et sont ouverts sur les autres, c'est un des intérêts de l'Open Source. Voilà pourquoi il faut communiquer autour du projet avec les autres développeurs, chercher des idées, des conseils, les mailing lists, les forums, freenodes, serveurs RCS, etc. Et pour cela il est évidemment plus pratique de parler anglais.

4 - Documenter

Dernier conseil : écrire un minimum de documentation pour rendre le projet accessible...

code existant. Pourtant, comme l'indique Julien Ballet, « *L'incitation à travailler sur le code d'autres personnes est formateur !* ». C'est en la matière une problématique sensible pour les éditeurs à la recherche de gains de productivité et d'efficacité. Car il existe un réel intérêt - « *une sensibilité croissante* » - des éditeurs pour les individus

capables de développer de nouvelles solutions émergentes. A condition de réduire voire supprimer leur besoin de développer afin de se concentrer sur les services qu'ils pourront ajouter et sur lesquels ils gagneront de l'argent...

Yves Grandmontagne

Un stage pour un futur jeune diplômé

Créé par quatre jeunes diplômés, AJstage se propose de mettre en relation les étudiants à la recherche d'un stage et les entreprises à la recherche de stagiaires.

Que l'on soit étudiant ou entreprise, que l'on recherche ou que l'on propose un stage, le recrutement des stagiaires est généralement une épreuve. L'étudiant prend position et soumet sa demande à des entreprises souvent sans savoir si ces dernières recrutent ou si son profil peut correspondre. Et du côté de l'entreprise les candidatures spontanées s'accumulent et leur traitement est bien souvent aléatoire, avec parfois en conclusion une inadéquation entre ses attentes et le stagiaire qui se présente.

Quatre jeunes diplômés issus d'universités, d'écoles d'ingénieur et de management - où

The screenshot shows the AJstage website interface. At the top, there's a navigation bar with 'STAGE' and 'AJstage - Stage, Apprentissage, VIE, CDI'. Below this, there are two main sections: 'Espace étudiant' (Student Space) and 'Espace recruteur' (Recruiter Space). The 'Espace étudiant' section has a login form with fields for 'Nom', 'Prénom', 'Email', and 'Mot de passe', and a 'Connexion' button. The 'Espace recruteur' section has a 'Le recrutement étudiant sur mesure' button. On the left, there's a sidebar with 'Outils' (Tools) and 'Aide' (Help) sections. The 'Outils' section includes links to 'Stage', 'VIE', 'Apprentissage', and 'CDI'. The 'Aide' section includes links to 'Maître service', 'Mes clients', 'Mes partenaires', and 'Mes équipes'. The 'Aide' section also includes a 'Contact' section with links to 'Conseils CV / LM', 'Conseils CV', 'Conseils entretien', and 'Conseils pour la mission'. The 'Aide' section also includes a 'Orientation' section with links to 'Téléphones', 'Fiches Métiers', 'Liens utiles', and 'FAQ des stagiaires'. The main content area shows a table of 'Dernières offres' (Latest offers) with columns for 'Entreprise', 'Intitulé', 'Début', and 'Durée'. The table lists several offers from companies like NIZI, Charge des partenariats internationaux, Charge des partenariats, Développeur .NET, Business Development - Recherche terrain, Design/Marketing, and Commercial start-up internet. On the right, there's a section 'RCS' (Réseau de Compétences) with a 'Travelavanc...' button and a 'picuous' button. At the bottom, there's a section 'e-reflex' with a 'Vos tous nos clients' button.

ils ont probablement rencontré les mêmes difficultés ! - ont créé leur start-up, AJstage, dans le but de proposer une solution pour fluidifier la relation étudiant-entreprise. L'en-

treprise dépose sa demande, stage, apprentissage, VIE et CDI. L'étudiant dépose profil, formation suivie, compétences, etc. AJstage dispose d'un moteur composé d'algorithmes de matching qui trie et assure la mise en relation des deux protagonistes [étudiant-entreprise] qui répondent le mieux à leurs intérêts.

Le modèle économique d'AJstage est simple : gratuit pour l'étudiant, qui peut également bénéficier de services de conseils personnalisés ; l'entreprise ne règle que le recrutement effectué. Le service semble pertinent, AJstage a déjà séduit des entreprises comme Dexia, Manpower ou Nestlé. www.ajstage.com

TENDANCES DE L'EMPLOI INFORMATIQUE

Le marché de l'emploi informatique recèle un paradoxe : de nombreuses entreprises cherchent à recruter, mais peinent à trouver des candidats susceptibles de satisfaire leur demande; mais dans le même temps les dizaines de milliers d'informaticiens (environ 5% de la profession) sont au chômage. La principale difficulté n'est pas de faire correspondre un poste et un demandeur, mais de trouver le bon profil... Concrètement, certains postes sont plus difficiles à pourvoir que d'autres !

Une étude de la division Network Management du spécialiste de la gestion de réseaux Ipswitch révèle quels sont actuellement les postes informatiques les plus difficiles à pourvoir.

Il apparaît, tout d'abord que les entreprises souffrent d'une pénurie de compétences sur le support réseau, alors que cette fonction est aujourd'hui essentielle pour assurer la haute disponibilité des services des réseaux et la continuité d'activité. 27,9% des postes actuellement les plus difficiles à pourvoir concernent l'Administrateur système. Suivent le Responsable réseau (22,7%), le

Responsable informatique (17,2%), le Développeur logiciel (14,6%), l'Analyste programmeur (10,4%), et enfin le Responsable télécom (7,1%).

> Ces entreprises qui recrutent

Les grands acteurs du secteur et les SSII continuent de recruter au rythme de leurs besoins, ou plus précisément des contrats engagés et du turnover. Pas de grands mouvements, juste du suivi ou des besoins ponctuels, en attendant de voir la tendance qui s'affichera durant les premières semaines de 2012. Dans ces conditions, Programmez ! vous invite à consulter les sites spécialisés afin de repérer les annonces de recrutements ponctuels.

> Coup de projecteur sur Bull

Le géant français des systèmes d'information maintient le rythme de ses recrutements : après avoir accueilli environ 1000 collaborateurs dans le monde en 2011, Bull vient d'annoncer qu'il recrutera de nouveau 1000 collaborateurs en 2012. La moitié de ces recrutements concerne la France, 500 autres postes concerneront le reste du monde, dont environ 150 pour la Pologne et 100 au Brésil.

> Quel est le profil des recrutements ?

Bull reçoit près de 10 000 candidatures spontanées par an en France. 25% des postes à pourvoir sont plus particulièrement destinés à des jeunes diplômés issus d'écoles d'ingénieurs ou de cursus universitaires. Un autre 25% des postes est pourvu par cooptation ou par réseau.

> Bull recrute pour quels métiers ?

Bull recrute régulièrement des experts dans le développement logiciel (Java J2EE, Décisionnel, ERP/SAP, etc.), des experts dans les technologies Bull (HPC, Cloud, Stockage/Archivage, etc.), des Architectes SI, des chefs de projet ou des experts sécurité. La maîtrise d'un secteur d'activité (secteur public, défense, télécoms, énergie, etc.) est un atout supplémentaire.

> De l'apprentissage et des stages :

Bull offre également de nombreuses opportunités pour accompagner les étudiants des écoles et universités avec lesquelles il collabore : le groupe recrute chaque année plus de 150 apprentis et accueille plus de 20 stagiaires. Une partie d'entre eux se voit proposer un poste à l'issue de la formation. www.bull.fr

ABONNEMENT

PDF

30 € par an

soit **2,73 € le**

numéro

www.programmez.com



**Abonnement
INTÉGRAL**

Pour un supplément de 10 € an

**accès illimité
aux archives**

Cette option est réservée aux abonnés pour 1 an au magazine, quel que soit le type d'abonnement (Standard, Numérique, Etudiant). Le prix de leur abonnement normal est majoré de 10 € (prix identique pour toutes zones géographiques). Pendant la durée de leur abonnement, ils ont ainsi accès, en supplément, à tous les anciens numéros et articles/ dossiers parus.

Installation de **CUDA C** avec Visual Studio 2010

2^e partie

Dans la première partie de cette série d'articles consacrés à l'introduction à la programmation GPU (Graphics Processing Unit), nous avons présenté les différentes technologies et l'algorithme qui sera utilisé pour chacune d'elles. Nous avons constaté les performances de cet algorithme, à la fois en mode séquentiel et en mode parallèle sur un CPU (Central Processing Unit) de type quad-cœurs.

Dans cette seconde partie, nous allons nous concentrer sur l'installation du Framework **nVIDIA CUDA** (*Compute Device Unified Architecture*) et de son intégration avec *Visual Studio 2010*.

> Vérification de votre carte graphique

Par nature, la programmation **nVIDIA CUDA** exige une carte graphique **nVIDIA** compatible **CUDA**. Vous trouverez à l'adresse suivante, l'ensemble des cartes **nVIDIA** compatibles **CUDA** : <http://developer.nvidia.com/cuda-gpus> [Fig.1].

Les cartes sont classées par catégorie de produits. L'offre est extrêmement diverse puisque les cartes GPU couvrent un territoire allant de machines type « *data-center* » jusqu'aux PC portables. La colonne « *Compute Capability* » indique la version de l'architecture matérielle CUDA relative à la carte associée. La version 2.0 est actuellement la dernière version, cependant vis-à-vis de notre étude qui est orientée découverte de la programmation GPU, la version de l'architecture n'a pas d'importance. Si vous ne possédez pas une carte **nVIDIA** compatible **CUDA**, vous ne pourrez malheureusement pas exécuter du code **CUDA** (le mode émulation de la carte graphique n'est plus supporté en **CUDA C 4.0**). Si au contraire vous avez une carte compatible **CUDA**, vous pouvez passer au téléchargement des produits.

> Télécharger les produits

Si votre machine s'exécute sur Windows 7, Windows Vista ou Windows XP, vous trouverez tous les éléments à télécharger à cette adresse : <http://developer.nvidia.com/cuda-toolkit-40#windows> [Fig.2].

Il est préférable d'installer les différents packages dans l'ordre présenté sur la page Web. J'ai donc repris les cinq étapes correspon-

dant à une installation complète, comprenant le **driver**, le **Toolkit de développement**, le **SDK complémentaire** et l'**outillage de mise au point**.

La première étape consiste à installer le **driver** à la fois adapté à votre carte vidéo et à votre système d'exploitation (32bit ou 64bit). Les nouvelles machines équipées d'une carte **nVIDIA** disposent déjà d'un driver **nVIDIA** à jour, mais si vous aviez un doute, vous pouvez toujours vérifier et le cas échéant installer la dernière version si votre version n'est pas à jour.

L'étape essentielle repose sur le **CUDA Toolkit**. En effet, ce package comprend tous les éléments pour développer : compilateur, profileur et un ensemble de bibliothèques complémentaires [Fig.3].

N.B. : Je vous invite à télécharger aussi la **documentation** respective aux différents packages que vous souhaitez installer.

Malheureusement pour les développeurs **Visual Studio 2010**, le package **CUDA Toolkit** présente un petit problème au niveau du fichier **MSBUILD** permettant la compilation et l'édition de lien **CUDA C** dans **Visual Studio**. Il faut donc télécharger aussi le package « **NEW* CUDA Toolkit 4.0 Build Customization BUG FIX Update Fixes error message « \$CUDABuildTasksPath property is not valide* », afin d'obtenir une compilation **CUDA C** opérationnelle [Fig.4].

Après avoir téléchargé ce ZIP dans un répertoire temporaire, vous trouverez le fichier « **README_CUDA4-0-BuildCustomization.txt** » qui vous fournira les instructions d'installation. Mais en deux mots vous devrez copier le fichier « **CUDA 4.0.targets** » et « **CUDA 4.0.props** » en fonction de votre système d'exploitation :

- **32bit OS:** C:\Program Files\MSBuild\Microsoft.Cpp\v4.0\BuildCustomizations
- **64 bit OS:** C:\Program Files (x86)\MSBuild\Microsoft.Cpp\v4.0\BuildCustomizations

CUDA-Enabled Tesla GPU Computing Products

Tesla Workstation Products		Tesla Data Center Products	
GPU	Compute Capability	GPU	Compute Capability
Tesla C2050/C2070	2.0	Tesla M2050/M2070/M2090	2.0
Tesla C1060	1.3	Tesla S2050	2.0
Tesla C870	1.0	Tesla S1070	1.3
Tesla D870	1.0	Tesla M1060	1.3
		Tesla S870	1.0

CUDA-Enabled Quadro Products

Quadro Desktop Products		Quadro Mobile Products	
GPU	Compute Capability	GPU	Compute Capability
Quadro 6000	2.0	Quadro 5010M	2.0
Quadro 5000	2.0	Quadro 5000M	2.0
Quadro 4000	2.0	Quadro 4000M	2.0
Quadro 4000 for Mac	2.0	Quadro 3000M	2.0
Quadro 2000	2.0	Quadro 2000M	2.0
Quadro 2000D	2.0	Quadro 1000M	2.0
Quadro 600	2.0	Quadro FX 3800M	1.1

WINDOWS 7, VISTA, WINDOWS XP

DOWNLOADS	Fig.2
Developer Drivers for WinXP (270.81) Support for XP on notebooks is being phased out and is not available for this release. See Release Notes and Getting Started Guides for more information.	32-bit 64-bit
Developer Drivers for WinVista and Win7 (275.33)	32-bit 64-bit
Notebook Developer Drivers for WinVista and Win7 (275.33)	32-bit 64-bit
CUDA Toolkit • C/C++ compiler • Visual Profiler • GPU-accelerated BLAS library • GPU-accelerated FFT library • GPU-accelerated Sparse Matrix library • GPU-accelerated RNG library • Additional tools and documentation	32-bit 64-bit documentation
NEW CUDA Toolkit 4.0 Build Customization BUG FIX Update Fixes error message "\$CUDABuildTasksPath property is not valid"	download
GPU Computing SDK - complete package including all code samples	32-bit 64-bit browse online
Parallel Hopt 2.0	download
Learn about additional tools, libraries, and more...	CUDA Ecosystem
CUDA Tools SDK (APIs for 3rd party performance analysis tools and cluster management solutions)	32-bit 64-bit



Installation du **NVIDIA CUDA Toolkit v4.0**

Vous pouvez aussi installer le **GPU Computing SDK** qui arrive avec des bibliothèques complémentaires et de nombreux exemples déclinés dans diverses technologies : **CUDA C**, **Microsoft Direct Compute** et **OpenCL** [Fig.5].

Certaines démos sont très impressionnantes et elles ont le mérite d'être livrées avec leurs sources [Fig.6].

Par exemple l'application N-Body implémentée via Direct Compute. Vous trouverez cet exemple pour une version 64 bit : C:\ProgramData\NVIDIA Corporation\NVIDIA GPU Computing SDK 4.0\DirectCompute\bin\win64\Release\nBodyCS.exe

Si vous souhaitez diagnostiquer votre code **CUDA**, je vous invite à installer **Parallel Nsight 2.0**. Malheureusement, ce package est un peu contraignant, puisqu'il vous réclame une exécution sans le thème **Aero** et sans l'accélération matérielle **WPF** (des fichiers de type « .reg » sont fournis avec l'installation pour modifier la base de registre afin de désactiver ou d'activer l'accélération matérielle). Cependant, ce package est vraiment très important lorsque vous développez avec **CUDA C** sous Windows [Fig.7].

> Configuration de CUDA C dans Visual Studio 2010

Activer la reconnaissance des fichiers CUDA type « .cu »

Par défaut, Visual Studio ne reconnaît pas les fichiers **CUDA** dont l'extension est « .cu ». Vous devez donc personnaliser Visual Studio :

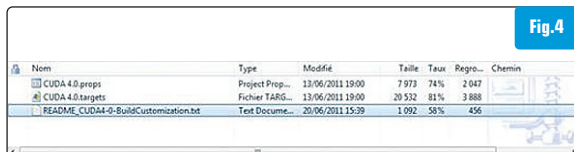
- Dans Visual Studio 2010, choisissez le menu « **Tools-> Options...** » [Fig.8].
- Ouvrez la section « **Text Editor** » et cliquez dans l'arborescence le nœud "File Extension". Tapez cu dans la zone texte « **Extension** » en choisissant au préalable « **Editor** » « **Microsoft Visual C++** », puis validez avec le bouton « **Apply** » afin d'enregistrer la modification. Ensuite, frappez « **Ok** » pour sortir [Fig.9].

> Ajouter la coloration syntaxique des extensions CUDA C

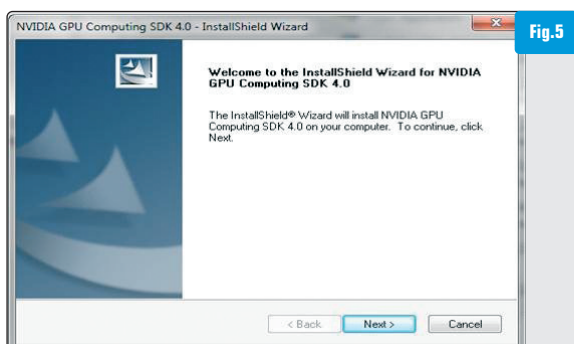
Lorsque vous développerez vos applications avec **CUDA C**, vous serez sans doute plus confortable si votre éditeur de code reconnaît les extensions **CUDA C**, ceci vous permettant de lire plus facilement le code **CUDA C**. Pour installer cette coloration syntaxique, il vous faut aller dans le répertoire : **C:\ProgramData\NVIDIA Corporation\NVIDIA GPU Computing SDK 4.0\doc\syntax_highlighting**. Vous y trouverez deux répertoires correspondant respectivement aux versions de Visual Studio 2005 et de Visual Studio 2008 [Fig.10]. Pour une installation avec Visual Studio 2010, vous pouvez choisir la version de Visual Studio 2008. Sur un système 64 bit vous devez vous rendre dans le répertoire "**C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE**" (répertoire "**C:\Program Files\Microsoft Visual Studio 10.0\Common7\IDE**" sur un système 32 bit), vous devriez trouver le fichier **usertype.dat**. Si ce fichier n'existe pas vous pouvez directement copier le fichier du package **CUDA**. Si le fichier est déjà présent vous pouvez éditer son contenu, puis copier ce contenu du fichier **usertype.dat** de la distribution **CUDA** à la fin du fichier "**C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE \usertype.dat**" déjà présent. Les mots clefs **CUDA C** apparaîtront dans la couleur bleue.

> Vérifier l'installation du runtime CUDA

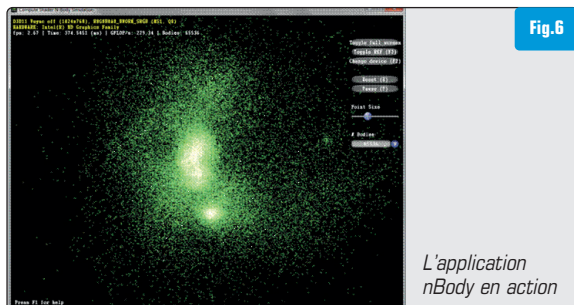
Vous pouvez lancer le **nVIDIA GPU Computing SDK 4.0**, dont une icône devrait être sur votre bureau (si vous n'avez pas décoché la case à l'installation). Vous devriez obtenir un écran similaire à l'image ci-dessous [Fig.11]. L'application est organisée en onglets thématiques : **CUDA C Code Samples**, **CUDA Libraries Samples**, **OpenCL Samples**, **Documentation**. Pour vérifier si votre installation est cor-



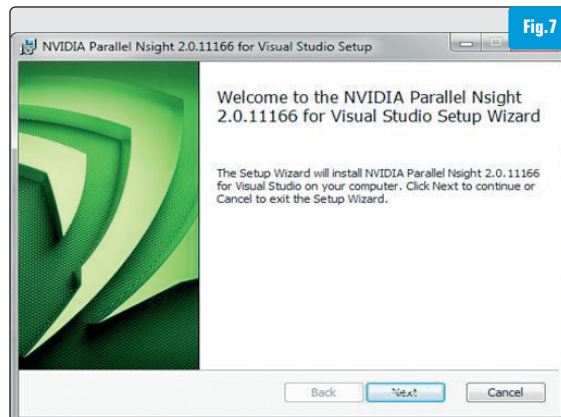
Distribution du package *NEW* CUDA Toolkit 4.0 Build Customization BUG FIX



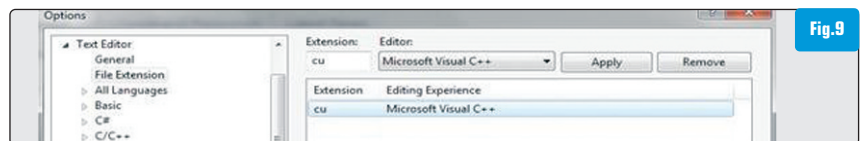
Installation du package NVIDIA GPU Computing SDK 4.0



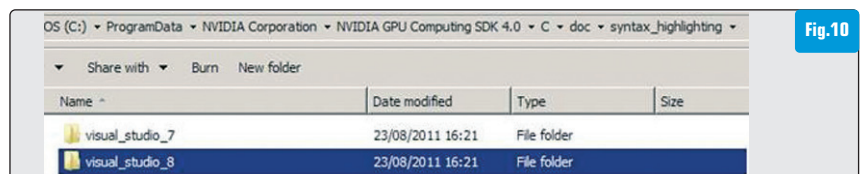
L'application nBody en action



Installation du package NVIDIA Parallel Nsight 2.0



Activer la reconnaissance des fichiers CUDA dans Visual C++



Packages pour activer la coloration syntaxique CUDA

recte et obtenir des informations sur la configuration de votre architecture **CUDA**. En tant que programmeur parallèle sur CPU, le nombre de cœurs disponibles vis-à-vis de votre calcul est un élément important. En effet, la rapidité de traitement dépend naturellement du nombre de threads engagés dans votre traitement. En programmation parallèle sur GPU, le nombre de threads disponibles est tout aussi important, si ce n'est plus. Si vous ne connaissez pas la capacité de votre carte graphique, vous pouvez lancer l'exécutable `deviceQuery.exe`, fourni dans la distribution du SDK en x64 : `C:\ProgramData\NVIDIA Corporation\NVIDIA GPU Computing SDK 4.0\bin\win64\Release\deviceQuery.exe` [Fig.12].

Les informations retournées `deviceQuery.exe` nous sont très précieuses, car elles nous indiquent les capacités physiques de notre carte graphique vis-à-vis de notre prochaine programmation en **CUDA C**. Nous y reviendrons un peu plus loin, lorsque nous détaillerons l'architecture **CUDA**. Si vous n'avez pas rencontré de problème jusqu'à présent, vous pouvez passer à l'étape suivante : vérifier l'intégration de **CUDA C** au sein de Visual Studio.

> Création de notre premier programme CUDA C au sein de Visual Studio 2010

Cette étape va vous demander un petit effort de codage. Dans Visual Studio 2010, vous devez lancer l'assistant Visual C++ et choisir le patron Win32 puis sélectionner le type « **Win32 Console Application** » [Fig.13]. Valider les options par défaut et vous devriez obtenir un code C++ généré similaire.

Code généré par l'assistant

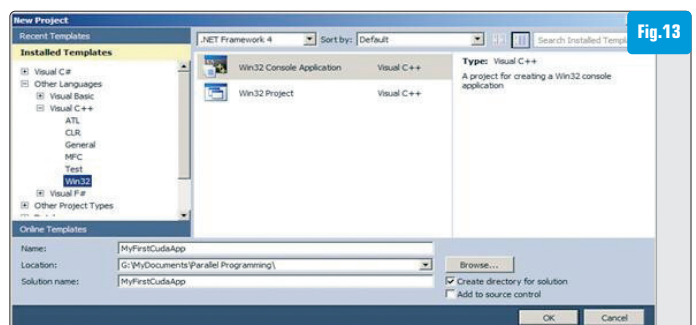
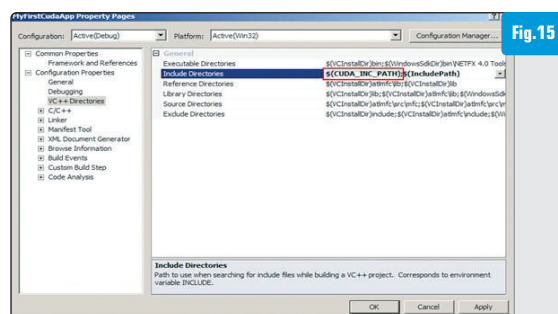
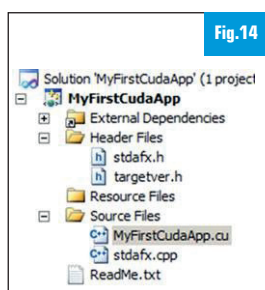
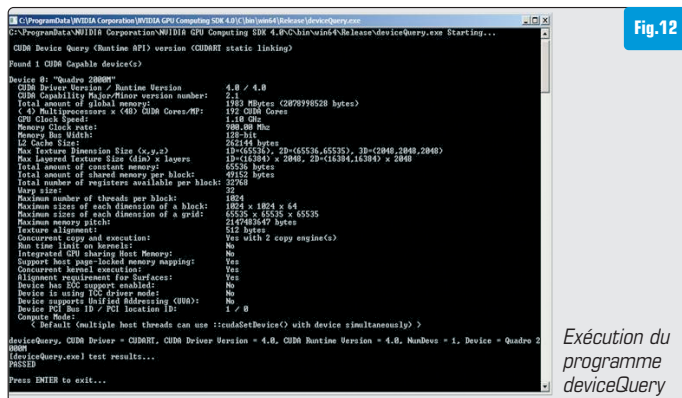
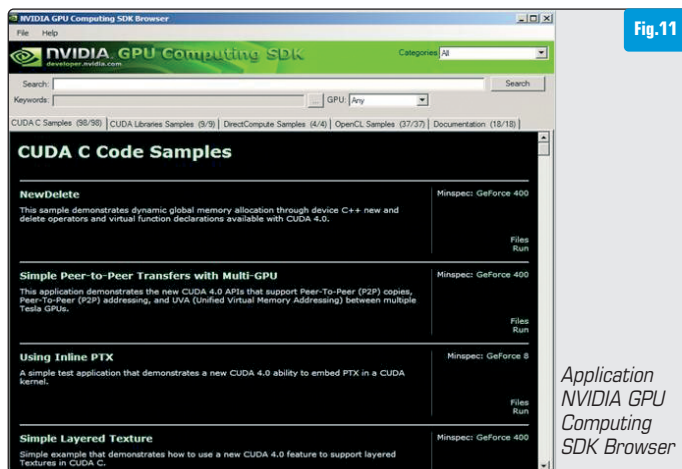
```
// MyFirstCudaApp.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
int _tmain(int argc, _TCHAR* argv[])
{
    return 0;
}
```

A ce stade, vous pouvez renommer le fichier `MyFirstCudaApp.cpp` en `MyFirstCudaApp.cu` [Fig.14].

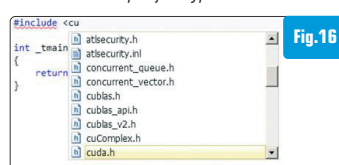
Pour que le compilateur C/C++ puisse trouver les fichiers headers de la distribution **CUDA**, vous devez paramétrer le chemin correspondant. Vous devez aller dans le menu projet, sélectionner les propriétés du projet. Dans le nœud « **Configuration Properties** », sélectionnez la ligne « **VC++ Directories** » [Fig.15].

Cette variable a été déclarée durant la phase d'installation. Vous pouvez vérifier que les fichiers d'en-tête sont bien accessibles depuis Visual Studio en ajoutant le fichier `cuda.h` [Fig.16].

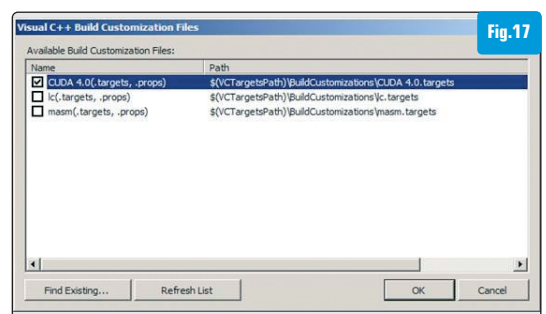
Vous pouvez alors activer la compilation **CUDA C**. Vous devez afficher la boîte de dialogue *Visual C++ Build Customization Files*. Vous



Création d'un projet type console avec Visual C++



Vérifier que les fichiers CUDA sont accessibles



Solution de notre premier projet CUDA

Ajouter la variable \$(CUDA_INC_PATH) à la racine des chemins d'inclusion

Activer l'utilisation de l'extension CUDA 4.0 pour la compilation du projet

pouvez par exemple faire un clic droit sur le projet de la fenêtre «Solution Explorer» et sélectionner le menu «**Build Customizations**...» [Fig.17].

Vous devez cocher la ligne **CUDA 4.0**. Une fois activée la compilation **CUDA**, vous devez sélectionner le fichier **MyFistCudaApp.cu** et afficher les propriétés du fichier [Fig.18].

Puis, vous devez choisir au niveau de la ligne «**Item Type : CUDA C/C++**». Enfin, au niveau du projet, vous pouvez saisir le chemin où se trouve le **CUDA Toolkit** via la variable **\$(CUDA_PATH)** [Fig.19].

Pour terminer le paramétrage **CUDA C**, il ne vous reste plus qu'à ajouter la librairie **CUDA**, **cuda.lib**, dans la partie édition de lien au niveau de la ligne «**Additional Dependencies**» [Fig.20].

Pour vérifier que notre intégration dans Visual Studio fonctionne bien, je vous propose d'interroger votre carte graphique sur ses capacités de traitements. Nous allons donc reproduire partiellement le programme présenté précédemment, **deviceQuery.exe**. Pour réaliser cette petite vérification nous allons directement inclure dans le fichier principal de notre projet, les deux fichiers **cuda.h** (fichier essentiel pour le développement **CUDA**) et **cuda_runtime_api.h** qui contient la déclaration de la fonction **cudaGetDeviceProperties** dont la motivation est de retourner une structure **cudaDeviceProp** contenant les caractéristiques de votre carte.

Listing pour interroger votre code graphique

```
// MyFirstCudaApp.cu : Defines the entry point for the console
// application.
//
#include "stdafx.h"
#include <cuda.h>
#include <cuda_runtime_api.h>
int _tmain(int argc, _TCHAR* argv[])
{
    cudaDeviceProp prop;
    cudaGetDeviceProperties( &prop, 0);
    printf( "Information for device %s\n\n", prop.name);
    printf( "Maximum number of threads per block: %d\n",
        prop.maxThreadsPerBlock );
    printf( "Maximum size of each dimension of a block: (%d, %d, %d)\n",
        prop.maxThreadsDim[0], prop.maxThreadsDim[1],
        prop.maxThreadsDim[2] );
    printf( "Maximum size of each dimension of a grid: (%d, %d, %d)\n",
```

```
prop.maxGridSize[0], prop.maxGridSize[1],
prop.maxGridSize[2] );
printf( "\n" );
return 0;
}
```

Vous pouvez passer à la compilation où vous observerez une compilation hybride entre le compilateur Microsoft C/C++ et le compilateur **CUDA C** (**nvcc.exe**) [Fig.21].

Si la compilation est un succès, il ne vous reste plus qu'à vérifier l'exécution de notre premier programme **CUDA** [Fig.22].

Le programme affiche les trois propriétés de ma carte graphique réclamées par notre programme. On note que les valeurs sont en adéquation avec le programme **deviceQuery.exe**. Arrivé à ce stade, vous disposez d'un environnement de développement d'applications **CUDA** opérationnel.

> En conclusion

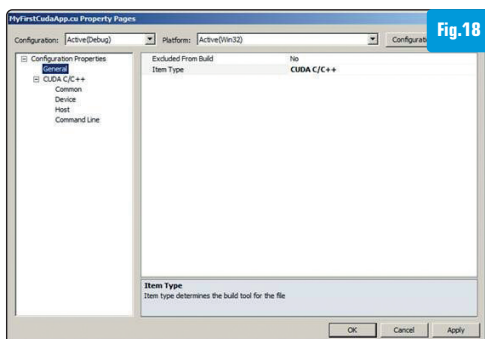
L'installation du Framework **CUDA C 4.0** avec Visual Studio 2010 exige un peu d'attention. Si vous souhaitez créer de nombreux projets **CUDA**, vous pouvez utiliser un fichier **Property Sheets (C++)** afin de ne pas ressaisir tout le paramétrage du projet à chaque fois: <http://msdn.microsoft.com/en-us/library/a4xbdz1e.aspx>. Après avoir configuré **CUDA C** dans l'environnement Microsoft, vous devriez retrouver un confort assez similaire à un projet C/C++ standard. Enfin, nous avons mis l'accent sur des informations clefs de l'architecture **CUDA**, comme le nombre de threads par bloc ou la dimension des blocs sans pour autant expliquer pourquoi ces valeurs étaient si importantes.

Dans la prochaine partie, nous détaillerons l'organisation des threads **CUDA**, puis nous montrerons comment dimensionner vos calculs en **CUDA C** afin d'obtenir des traitements rapides et justes. Enfin, nous implémenterons l'algorithme présenté dans la première partie, puis nous constaterons l'expressivité du code et les performances obtenues.

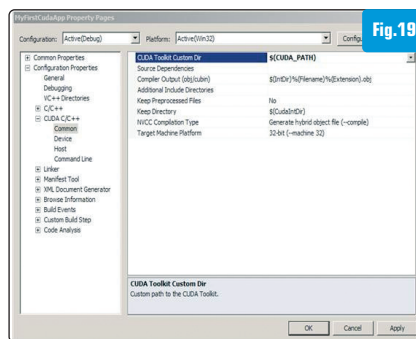


Bruno Boucard

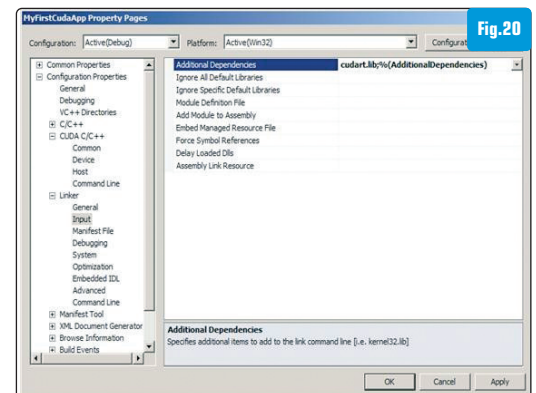
Indépendant, expert en parallélisme. MVP Technical Computing
<http://blogs.msdn.com/b/devpara/>



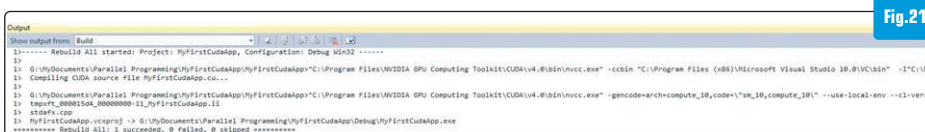
Configurer votre type de fichier en CUDA C/C++



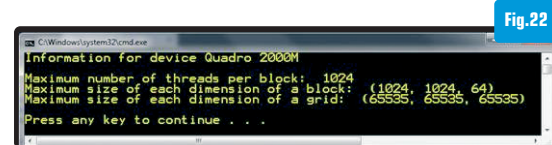
Indiquer où se trouve le CUDA Toolkit



Indiquer la librairie **cuda.lib** dans la liste des librairies à inclure dans l'édition de lien.



Compilation C/C++ avec le compilateur **CUDA**



Exécution de notre premier programme **CUDA**

Accéder aux données SharePoint avec REST & JSON

Les web services ont révolutionné le monde du web en permettant à des applications hétérogènes de communiquer entre elles et d'échanger des informations. Pour mettre en place cette technologie, plusieurs moyens et architectures sont disponibles qui permettent de réaliser des appels distants ou de consommer des ressources.

XML-RPC : lancé par la société « Userland Software », cette technologie assure l'appel des procédures distantes (Remote Procedure Call) à travers la norme XML pour le codage des données. La consommation du service se fait via le protocole http qui transporte les données. Ce protocole est l'ancêtre de SOAP.

SOAP : Créé par Microsoft (norme industrielle) et soutenu par le W3C, ce protocole assure le même fonctionnement que XML-RPC. Toutefois, il est conseillé dans le cas d'échange de documents complexes (de taille importante). SOAP repose sur plusieurs protocoles comme http, SMTP, FTP ... Le format d'échange des flux de données via le protocole http est toujours du XML, auquel SOAP rajoute sa propre structure. En effet, un message SOAP contient un en-tête et une enveloppe qui encapsulent les données. La pile web service soap officielle du W3C est : http, SOAP, WSDL, UDDI.

REST : n'est pas un protocole mais plutôt un style d'architecture client/serveur. Ce style permet de consommer essentiellement des ressources à distance en utilisant le protocole http. A travers une URI qui est de syntaxe universelle, on peut effectuer des opérations de type CRUD (Create, Read, Update and Delete) tout en se basant sur les opérations http : Get, POST, PUT, DELETE, HEAD ou OPTION. Cette architecture est très proche du protocole http, ce qui rend son utilisation très facile.

Afin de comprendre mieux la différence entre ces modèles, nous dressons ici un tableau comparatif entre SOAP et REST. Le protocole XML-RPC ressemble beaucoup à SOAP et donc nous ne l'incluons pas dans notre comparaison :

Architecture	SOAP	REST
Critère		
Mise en place et développement	Nécessite des outils et Toolkits	Très facile à mettre en place
Gestion des erreurs	Pris en compte dans les enveloppes SOAP	Ne sont pas pris en compte
Résultat	XML	XML/JSON (moins verbeux)
Typage	Fort, le client doit connaître les types des données échangées	Faible
Sécurité	http(s), WS-* Standards	http(s), SSL
Protocoles utilisés	HTTP, SMTP, FTP,...	HTTP
Transfert de pièces jointes	Supporte l'envoi de fichiers	N'est pas très adapté
Appel côté client	Le parsing XML avec JavaScript est lourd	JSON reconnu directement par JavaScript (pas de parseur)

Les deux façons d'implémenter les deux services se basent essentiellement sur le langage XML. Le développeur doit analyser ses besoins afin de faire le bon choix en prenant en compte le type d'opé-

ration qu'il souhaite implémenter et le niveau de sécurité de son application, etc. Dans la suite de cet article, nous utiliserons l'architecture REST qui est déjà prise en compte dans SharePoint 2010.

> Architecture REST dans SharePoint 2010 :

SharePoint 2010 offre plusieurs façons pour accéder aux données d'une liste :

- Modèle objet serveur : utilisé lorsque vous développez sur le serveur SharePoint.
- Modèle objet client : utilisé lorsque vous exécutez votre code sur une machine distante.
- Web Services
- REST

Dans ce qui suit, nous allons nous intéresser au modèle objet client et en particulier à l'utilisation de l'architecture REST pour interroger vos données. SharePoint 2010 offre des web services WCF (Framework 3.5) pour accéder aux données d'une liste via l'architecture REST. Pour tester cette nouvelle fonctionnalité il faut ouvrir le web service **listdata** à travers l'url suivante :

http://nomserveur/_vti_bin/listdata.svc

NB : ADO.NET Data Services est installé par les pré-requis de SharePoint. Par contre, si l'OS du serveur est 2008 R2, ça ne fonctionne pas sans avoir installé un hotfix ou le SP1.

Pour la suite de l'article nous aurons besoin d'une liste que nous nommerons « Produits » :

- Produits : Titre, Description

Remplissez la liste pour avoir du contenu [Fig.1].

> Structure des URL pour accéder aux données

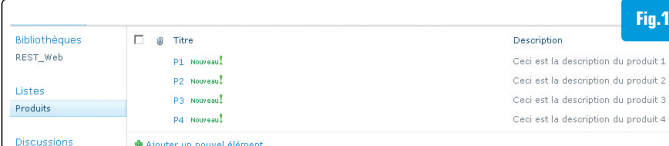
La syntaxe de base d'une url d'accès aux données est la suivante :

```
http://host/<service>/<EntitySet>[(<Key>)]/<NavigationProperty>[(<Key>)/...]]
```

Les éléments importants dans cette url sont :

- Le **data service** URI : c'est le nom de votre serveur suivi du nom de web service à consommer.
- L'**entity set** (optionnel): sert à mentionner le nom de l'entité que l'on souhaite retourner.

Exemple : `/Produits` permet de retourner tous les éléments de la liste produit.



Titre	Description
P1 Nouveau!	Ceci est la description du produit 1
P2 Nouveau!	Ceci est la description du produit 2
P3 Nouveau!	Ceci est la description du produit 3
P4 Nouveau!	Ceci est la description du produit 4

Fig.1

- La navigation property : peut être placée après l'entity set (séparé par « / ») indiquant le nom de la colonne que l'on souhaite retourner. Exemple : `/Produits(1)/Description`. Ici, on souhaite connaître la description du produit « P1 » qui est une clé primaire. Plusieurs options existent pour faire des jointures entre les entités, limiter le nombre de résultats retournés, etc.

Vous trouverez plus d'informations ici :

<http://msdn.microsoft.com/en-us/library/cc907912.aspx>

Voici ce que vous obtenez si vous essayez d'accéder à la liste « Produits » http://nomserveur/_vti_bin/listdata.svc/Produits [Fig.2]

Le résultat retourné par cette requête est un flux XML (au format Atom). Pour visualiser ce flux, il suffit d'aller dans Internet Explorer -> Tools -> Internet Options -> Content -> Feeds and Web Slices -> Settings -> Décocher « Turn on feed reading view ». Enfin, rechargez la page et vous aurez votre flux XML : [Fig.3].

> REST avec une application Web

Nous allons maintenant développer une Web Part qui liste les données de la liste SharePoint « Produits » en utilisant JavaScript.

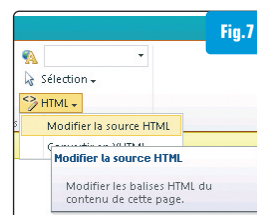
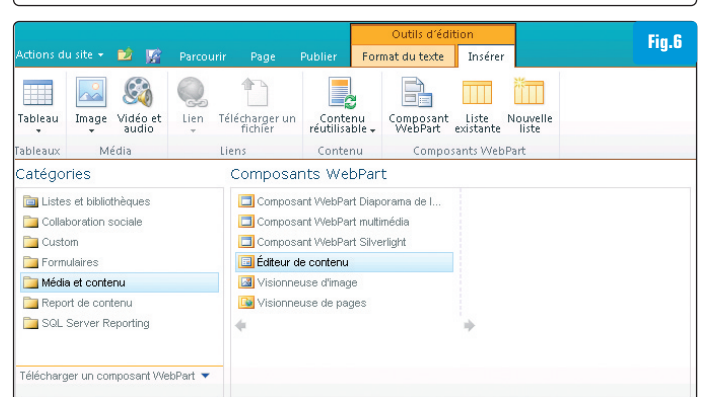
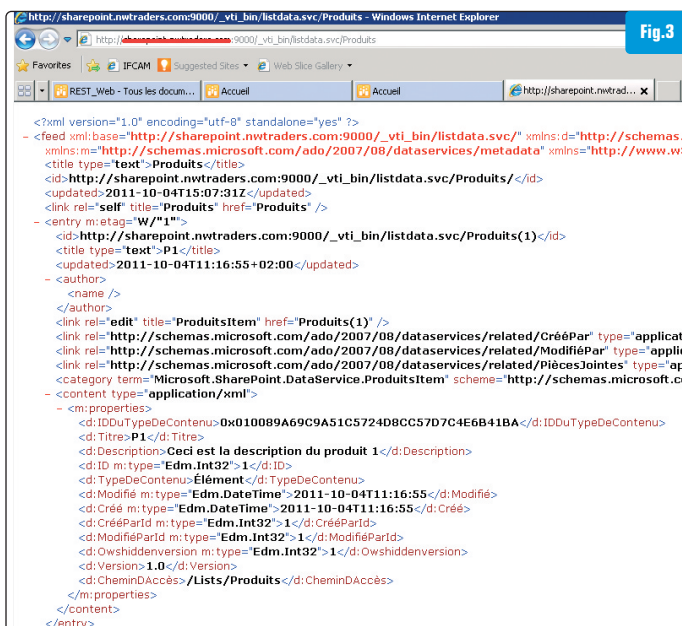
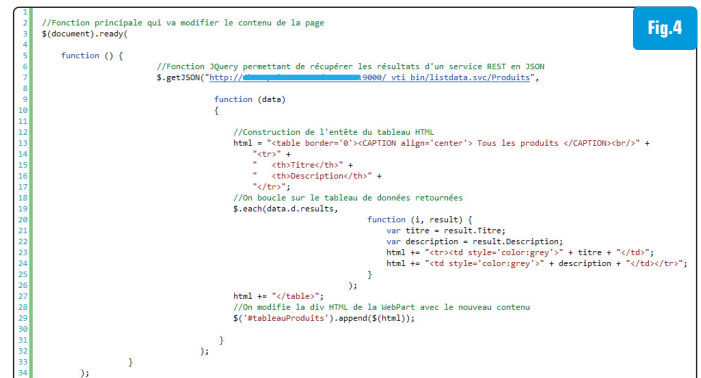
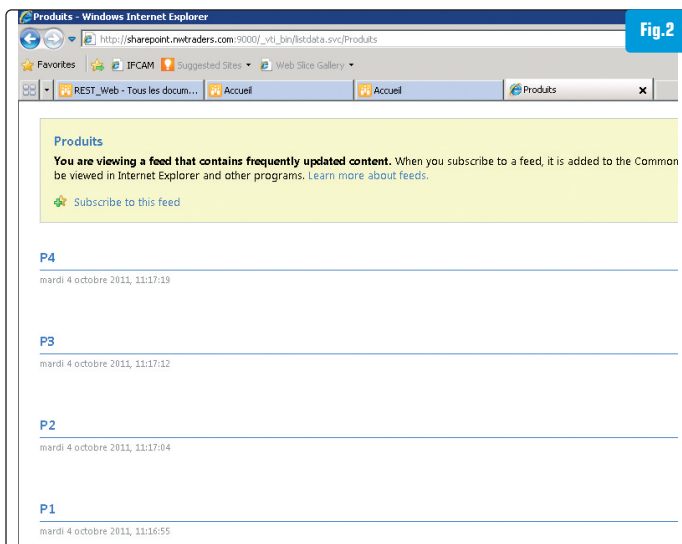
Nous n'allons pas utiliser le langage XML pour gérer les données retournées par le flux http mais plutôt le format JSON (JavaScript

Object Notation) qui fournit un support simple et léger pour communiquer avec le serveur. Pour ce faire :

- Ouvrez votre site dans un navigateur.
- Créez une nouvelle bibliothèque de documents « REST_Web » : Elle contiendra les fichiers JavaScript de notre exemple.
- Téléchargez la librairie JQuery et ajoutez-la à cette librairie.
- Créez un nouveau fichier avec un éditeur de texte « REST_Web.js » qui contiendra le script permettant l'accès à SharePoint.
- Rajoutez le code suivant dans ce fichier :
- Sauvegardez votre fichier et ajoutez-le à la bibliothèque de documents « REST_Web » [Fig.5].
- Allez sur une page du site et cliquez sur « Modifier la page ».
- Insérez une Web Part « Editeur de contenu » : Nous mettrons le code HTML dedans [Fig.6].
- Cliquez sur modifier la source HTML : [Fig.7].
- Insérez le code suivant : [Fig.8].
- Sauvegardez la page et vous aurez la liste des produits : [Fig.9].

> Débogage des requêtes avec Fiddler

Fiddler est un outil qui permet d'analyser les requêtes http effectuées sur la machine. Dans notre exemple, nous aurons besoin de



voir comment fonctionne l'architecture REST via les appels GET.

- Lancez Fiddler.
- Ouvrez l'URL suivante : http://nomserveur/_vti_bin/listdata.svc/Produits [Fig.10]
- Analysez la requête : Il s'agit bien d'une requête GET sur un hôte qui est le serveur SharePoint, le résultat par défaut retourné est XML.
- Allez dans l'onglet « Request Builder » et ajoutez « accept : application/JSON » et ré-exécutez la requête : [Fig.11].
- Remarquez la différence du résultat retourné par rapport à la première requête. Ici, vous obtenez les données sous forme de liste « d » et « results », ce qui explique l'utilisation de la boucle sur `data.d.results` dans le fichier JavaScript [Fig.12].

> REST avec une application WinForms

Nous pouvons faire du REST avec n'importe quel type d'application, ici nous allons voir comment consommer un service REST avec une application WinForms. L'objectif est d'ajouter un nouvel élément à la liste des produits via la requête POST. Pour cela :

- Créez une nouvelle application WinForms « REST_WindowsForms »
- Renommez « Form1.cs » par « Produits_Form.cs »
- Ajoutez un service à ce projet (Add Service Reference), de la même manière que la consommation d'un service Asp.net ou WCF.
- Saisissez l'URI suivante http://nomserveur/_vti_bin/listdata.svc
- Renommez le service en « REST_ServiceReference » et là, Visual Studio va générer un proxy permettant d'appeler les différentes méthodes et propriétés de ce service.

- Dans votre formulaire, glissez un composant de type DataGridView que vous nommerez « dataGridView_Produits ».
- Ajoutez deux labels et deux zones de textes pour saisir le texte et la description du produit.
- Ajoutez un bouton qui permettra l'ajout d'un nouveau produit.
- Effectuez un double-clic sur votre Form et saisissez le code suivant : [Fig.13].
- Exécutez votre programme, vous devez voir le DataGridView rempli.
- Saisissez le titre et la description du nouveau produit et cliquez sur le bouton « Ajouter » : [Fig.14].

> Débogage des requêtes avec Fiddler

Dans cet exemple, nous sommes en train d'effectuer une requête POST (l'opération Create) qui permet de créer un nouveau produit.

- Visualisez l'en-tête de la requête : nous retrouvons bien les données saisies dans le formulaire.

Conclusion

A travers cet article, nous avons montré l'architecture REST dans SharePoint 2010. Après avoir effectué une étude comparative entre les différentes méthodes de consommation de Web Services, nous avons développé en deux exemples l'utilisation des web services ADO.Net fournis par SharePoint 2010 pour accéder à ses données.

Amine Zemzemi - Winwise

```
<html>
<head>

<script src="/REST_Web/jquery-1.6.4.min.js" type="text/javascript"></script>
<script src="/REST_Web/REST_Web.js" type="text/javascript"></script>

</head>
<body>
    <div id="tableauProduits">
    </div>
</body>
</html>
```

Fig.8

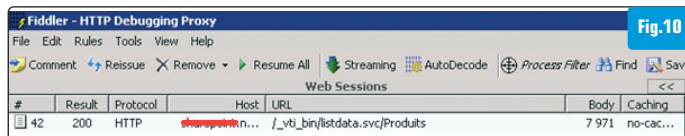


Fig.10

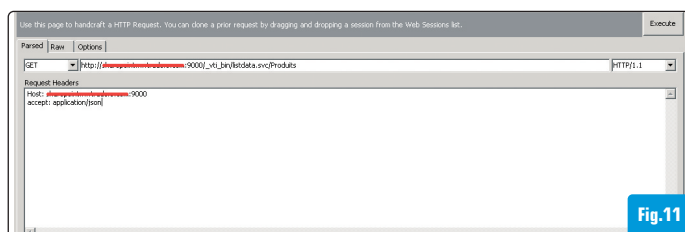


Fig.11

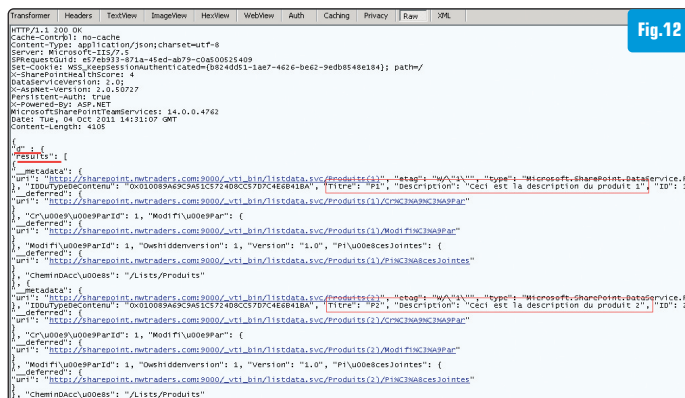


Fig.12

```
1 using System;
2 using System.Collections.Generic;
3 using System.ComponentModel;
4 using System.Data;
5 using System.Drawing;
6 using System.Linq;
7 using System.Text;
8 using System.Windows.Forms;
9 using System.Net;
10 using System.IO;
11 // Ajouter une référence au service ajouté
12 using REST_ServiceReference;
13
14 namespace REST_WindowsForms
15 {
16     public partial class Produits_Form : Form
17     {
18         //Objet du site SharePoint à s'y connecter "Voir appellation de votre site : 'NomSite'+DataContext
19         IncomingWailsDataContext _context;
20
21         public Produits_Form()
22         {
23             InitializeComponent();
24         }
25
26         private void Produits_Form_Load(object sender, EventArgs e)
27         {
28
29             //Connexion au site avec le proxy généré
30             string url = @"http://nomserveur:9000/_vti_bin/listdata.svc";
31             _context = new IncomingWailsDataContext(new Uri(url));
32             _context.Credentials = CredentialCache.DefaultCredentials;
33
34             //Génération du DataGridView
35             dataGridView_Produits.AutoGenerateColumns = false;
36             dataGridView_Produits.Columns.Add("Titre", "Titre");
37             dataGridView_Produits.Columns.Add("Description", "Description");
38             dataGridView_Produits.Columns[0].DataPropertyName = "Titre";
39             dataGridView_Produits.Columns[1].DataPropertyName = "Description";
40
41             //Chargement du grid
42             LoadData();
43         }
44     }
45 }
```

Fig.13

```
46 private void btnAjouter_Click(object sender, EventArgs e)
47 {
48     //Création d'un nouveau produit
49     ProduitsItem produit=new ProduitsItem();
50     produit.Titre = tbTitre.Text;
51     produit.Description = tbDescription.Text;
52     //Ajout du produit
53     _context.AddToProduits(produit);
54     //Sauvegarde des modifications
55     _context.SaveChanges();
56     //Rechargement du grid
57     LoadData();
58 }
59
60 public void LoadData()
61 {
62
63     //Récupération de tous les produits avec une requête LINQ
64     var res = from produit in _context.Produits
65               select produit;
66
67     dataGridView_Produits.DataSource = res.ToList();
68 }
69
70 }
71
72 }
```

Fig.14

Applications bureau riches en Silverlight avec le mode Out-Of-Browser

Silverlight est connu pour être la technologie de Microsoft créée pour développer des applications internet riches (Rich Internet Applications – RIA) et concurrencer la technologie Adobe Flash. Cette idée qui pouvait être valide lors des premières versions de Silverlight ne l'est désormais plus. En effet, depuis la version 3, et surtout depuis sa version 4, Silverlight s'est doté d'un mode Out-Of-Browser communément appelé OOB. Ce mode permet aux applications développées en Silverlight de s'exécuter hors du navigateur comme une application bureau (Rich Desktop Application - RDA) avec quelques différences.



Microsoft®
Silverlight™

Dans cet article nous allons faire un tour d'horizon des différentes fonctionnalités offertes par le mode OOB au travers d'une application d'exemple à destination des commerciaux de Contoso Inc. Cette application leur permettra de réagir plus rapidement aux demandes de leurs clients souhaitant louer la fameuse Dixel.

ACTIVATION DU MODE OUT OF BROWSER

Par défaut, une application Silverlight ne peut pas être lancée en mode OOB. Afin de pouvoir faire cela, il faut d'abord la configurer. Dans Visual Studio, il suffit de faire un clic droit sur le projet Silverlight et de cliquer sur « Propriétés ». Dans la fenêtre qui s'ouvre [Fig.1] il suffit de cocher la case « enable running application out of the browser » pour que celle-ci soit utilisable hors du navigateur.

L'application est maintenant installable sur le poste de l'utilisateur grâce à un clic droit lorsqu'elle s'exécute dans le navigateur [Fig.2]. Le processus d'installation demande alors à l'utilisateur où mettre les raccourcis vers l'application [Fig.3]. Une fois installée elle prend l'apparence d'une application bureau classique [Fig.4].

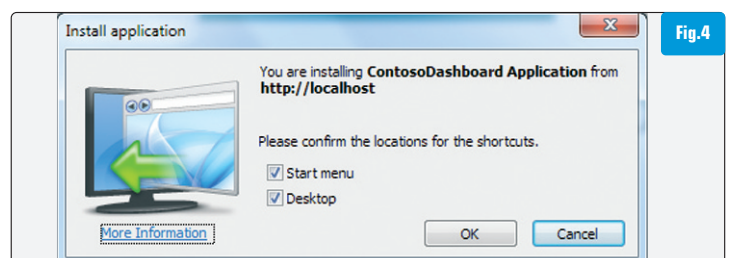
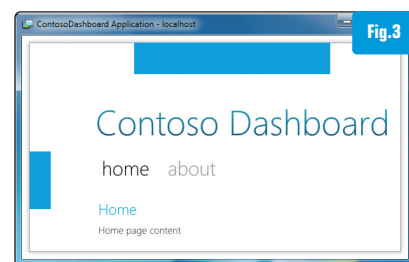
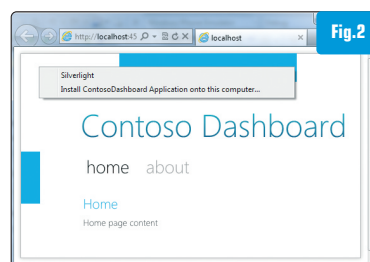
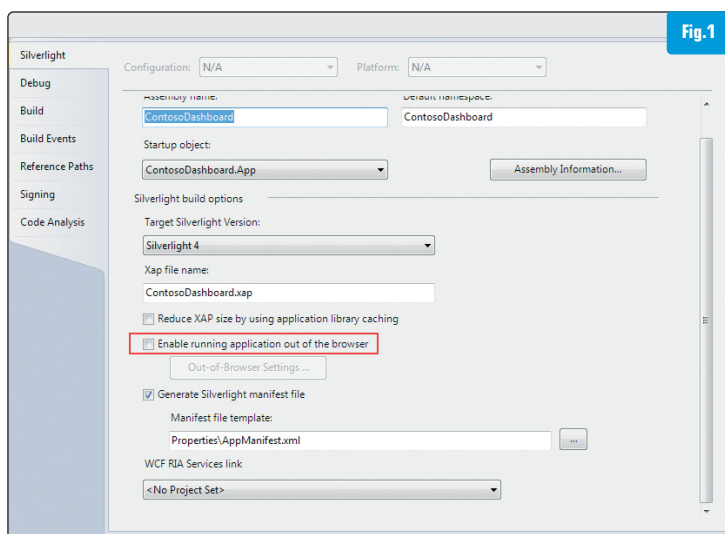
Pour désinstaller l'application il suffira ensuite de faire un clic droit lors de son exécution et de choisir de supprimer l'application dans le menu.

PERSONNALISATION DE L'EXPÉRIENCE D'INSTALLATION

Dans notre cas d'utilisation, nous avons décidé que notre application ne devrait s'exécuter qu'en mode OOB. Pour cela, nous allons créer un contrôle qui sera utilisé à la place de la page principale de l'application lorsqu'elle s'exécute dans le navigateur et qui proposera à l'utilisateur d'installer l'application. Dans l'évènement StartUp de l'application nous mettons le code suivant :

```
if (!IsRunningOutOfBrowser)
{
    this.RootVisual = new InstallationPage();
}
else
{
    this.RootVisual = new MainPage();
}
```

Lorsque l'utilisateur clique sur le bouton « installer » nous exécuterons le code suivant :



```
private void Button_Click(object sender, RoutedEventArgs e)
{
    if (App.Current.InstallState != InstallState.Installed)
        App.Current.Install();
}
```

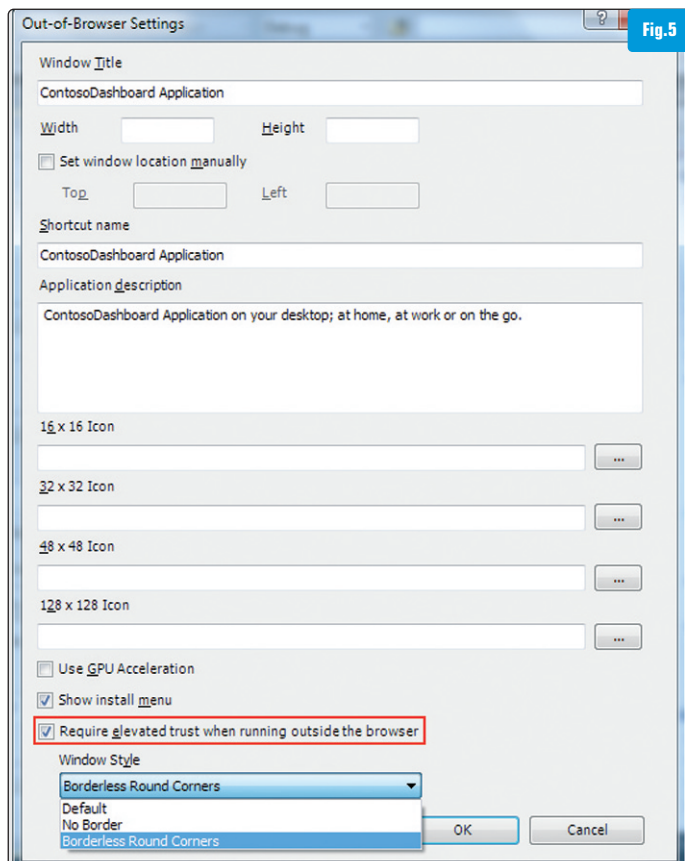
Une fois l'application installée elle se lancera automatiquement en mode OOB.

MODE DE CONFIANCE ÉLEVÉE

Une application Silverlight s'exécute dans un bac à sable qui l'isole du reste du système et ne lui fournit que des droits limités. Cependant en mode OOB, une application Silverlight peut demander à avoir accès à des droits supplémentaires qui vont lui permettre de fournir une expérience utilisateur plus riche grâce à une meilleure intégration dans le système. On dit alors que cette application fonctionne en mode de confiance élevée ou *elevated trust* en anglais (ne pas confondre avec une application Full-Trust comme en WPF par exemple).

Pour avoir accès à des droits supplémentaires il faut configurer notre application dans Visual Studio. Dans la fenêtre de propriétés du projet Silverlight, sous la case que nous avons cochée précédemment, se trouve un bouton « Out-of-Browser Settings ». En cliquant sur celui-ci nous avons accès à un certain nombre d'options telles que la taille de la fenêtre de l'application, les icônes utilisées ou bien le nom du raccourci. En bas de cet écran se trouve une case à cocher « Require elevated trust when running outside the browser ». En cochant cette case on peut définir si l'on souhaite conserver les bordures des fenêtres ou non et on obtient l'accès à des droits supplémentaires si l'utilisateur les accepte bien entendu [Fig.5].

Dès lors qu'une application est marquée pour fonctionner en mode



elevated trust, l'expérience d'installation diffère légèrement de précédemment. Si l'application n'est pas signée, l'utilisateur se verra présenter une fenêtre d'installation jaune le prévenant que l'éditeur de l'application n'a pas pu être vérifié [Fig.6]. Si l'application est signée avec un certificat valide alors la fenêtre d'installation sera déjà plus accueillante [Fig.7].

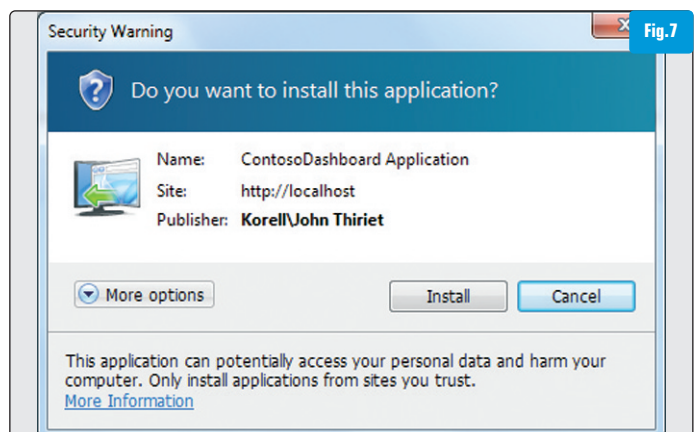
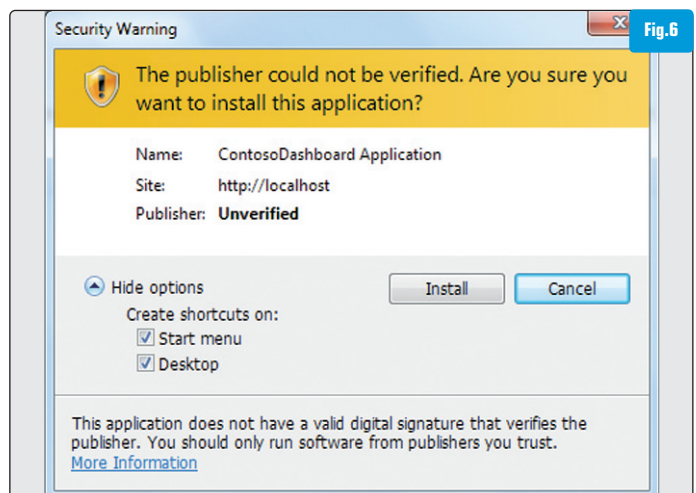
MISE À JOUR D'UNE APPLICATION OUT OF BROWSER

Silverlight propose un mécanisme de mise à jour d'une application Out Of Browser. Un petit détail à noter est que si l'application est en mode de confiance élevé, alors elle doit être signée avec un certificat valide pour que le mécanisme de mise à jour fonctionne.

Pour la mise à jour il suffit d'utiliser le code suivant :

```
Application.Current.CheckAndDownloadUpdateCompleted += Current
    _CheckAndDownloadUpdateCompleted;
Application.Current.CheckAndDownloadUpdateAsync();

...
private void Current_CheckAndDownloadUpdateCompleted(object sender,
    CheckAndDownloadUpdateCompletedEventArgs e)
{
    if (e.UpdateAvailable)
        MessageBox.Show("Update success.");
    else if (e.Error != null)
        MessageBox.Show("Update failed.");
}
```




```
else
    MessageBox.Show(«No available update.»);
}
```

La méthode `CheckAndDownloadUpdateAsync` lance la vérification d'une nouvelle version de l'application et la télécharge le cas échéant. Il peut arriver qu'une mise à jour échoue. C'est le cas lorsque la nouvelle version de l'application nécessite une nouvelle version de Silverlight ou que l'application passe d'un mode normal à un mode de confiance élevée. Dans le premier cas, la propriété `Error` contiendra une exception de type `PlatformNotSupportedException` et dans le second, une exception de type `SecurityException`. Dans les deux cas il faut inviter l'utilisateur à se rendre sur le site contenant l'application pour effectuer la mise à jour.

Si la mise à jour a été correctement effectuée, on notifiera l'utilisateur qu'un redémarrage de l'application est nécessaire pour qu'il travaille avec la dernière version.

UTILISATION DU RÉSEAU

Lorsqu'un client de Contoso demande des informations sur la Dixel, un commercial doit en être informé au plus vite. Notre application sera donc connectée en direct sur un serveur interne à Contoso qui nous informera d'un tel évènement. Pour cela, nous allons utiliser les sockets. Les sockets permettent une communication bidirectionnelle et très performante entre clients et serveurs.

Bien qu'il soit possible d'utiliser les sockets sans WCF nous avons opté pour ce dernier avec une liaison `netTcpBinding`, ce qui nous permet de bénéficier de la performance des sockets tout en ayant la facilité d'utilisation de WCF. Lorsqu'un message du serveur est reçu, les données des personnes à contacter doivent être ajoutées à une grille. Ajouter une référence à un service WCF utilisant les sockets se fait de la même façon qu'un service, la seule vraie différence se situe au niveau des endpoints WCF côté serveur [Fig.8].

La gestion des sockets n'est pas spécifique à Silverlight OOB, cependant une application n'étant pas en mode OOB et de confiance élevée ne peut établir de connexions qu'en utilisant des ports compris entre 4502 et 4534. Dans la plupart des cas, ces ports ne sont pas accessibles depuis l'extérieur d'une infrastructure informatique, ce qui limite généralement l'utilisation des sockets à des réseaux

internes. De plus, comme toute connexion réseau en Silverlight classique, dans le cas d'un appel inter-domaine, un fichier de stratégie de sécurité nommé `clientaccesspolicy.xml` [Fig.9] doit être présent sur le serveur afin d'autoriser explicitement les appels provenant d'autres domaines. En revanche, une application de confiance élevée n'est pas soumise à ces limitations.

LES TOASTS NOTIFICATIONS

Les notifications toast sont des petites fenêtres qui apparaissent en bas à droite du bureau [Fig.10]. Elles sont particulièrement utiles pour informer l'utilisateur qu'un évènement se produit lorsque l'application est réduite ou inactive. Voici comment créer une notification :

```
NotificationWindow notification = new NotificationWindow();
notification.Content = new TextBox { Text = «Hello world» };
```

Une fenêtre de notification est limitée à une taille de 400 pixels de large sur 100 de haut, elle ne peut pas être transparente ni avoir de bords arrondis, de plus, une seule d'entre elles peut être ouverte à la fois. Ces limitations ont été créées par souci de sécurité afin qu'une notification soit facilement identifiable par l'utilisateur comme telle.

LE CONTRÔLE WEBBROWSER

Le contrôle `webbrowser` nous permet d'intégrer une page internet dans une application Silverlight. Ce contrôle charge une instance d'Internet Explorer sur Windows et de Safari sur Mac OS pour afficher la page web souhaitée. La conséquence de cela est qu'il peut y avoir une différence de rendu de la page entre les différentes plateformes.

Dans notre application nous utiliserons le contrôle `webbrowser` pour afficher le site de Bing Maps en centrant la carte sur la ville de résidence de l'utilisateur.

Pour utiliser ce contrôle il faut tout d'abord créer une instance en XAML ou par code et ensuite appeler la méthode `Navigate` en passant en paramètre l'adresse de la page à afficher.

```
<WebBrowser x:Name=»browser» />
```

```
<?xml version="1.0"?>
<configuration>
  <system.serviceModel>
    <bindings>
      <netTcpBinding>
        <binding name="DuplexNetTcpBinding">
          <security mode="None"/>
        </binding>
      </netTcpBinding>
    </bindings>

    <behaviors>
      <serviceBehaviors>
        <behavior name="DuplexCommunicationNetTcpBehavior">
          <serviceMetadata httpGetEnabled="true"/>
          <serviceDebug includeExceptionDetailInFaults="true"/>
          <serviceThrottling maxConcurrentSessions="2147483647"/>
        </behavior>
      </serviceBehaviors>
    </behaviors>

    <services>
      <service behaviorConfiguration="DuplexCommunicationNetTcpBehavior" name="ContosoDashboard.Web.ContosoService">
        <host>
          <baseAddresses>
            <add baseAddress="net.tcp://localhost:4502"/>
            <add baseAddress="http://localhost"/>
          </baseAddresses>
        </host>
        <endpoint address="" binding="netTcpBinding" bindingConfiguration="DuplexNetTcpBinding" contract="ContosoDashboard.Web.IContosoService"/>
        <endpoint address="mex" binding="mexTcpBinding" contract="IMetadataExchange"/>
      </service>
    </services>

    <serviceHostingEnvironment multipleSiteBindingsEnabled="true" />
  </system.serviceModel>
</configuration>
```

Fig.8

```
<?xml version="1.0" encoding="utf-8"?>
<access-policy>
  <cross-domain-access>
    <policy>
      <allow-from>
        <domain-uri="*" />
      </allow-from>
      <grant-to>
        <socket-resource port="4502-4506" protocol="tcp" />
      </grant-to>
    </cross-domain-access>
  </access-policy>
```

Fig.9

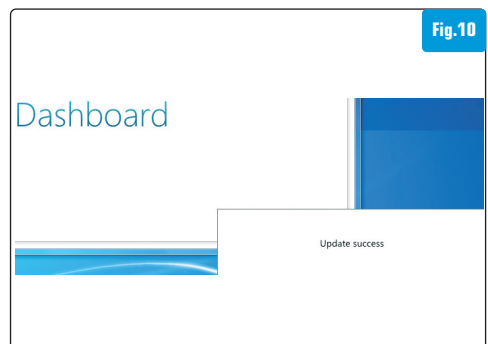


Fig.10

```
browser.Navigate(new Uri(«http://www.bing.com/maps/?where=» +
_customer.City));
```

COM AUTOMATION

En mode de confiance élevée, nous avons la possibilité d'utiliser les composants COM enregistrés sur la machine de l'utilisateur tels que ceux de la suite Office. Cette fonctionnalité n'est disponible que sur Windows en raison de l'absence de tels composants sur Mac OS.

Dans notre exemple, nous allons voir comment ouvrir la liste des clients souhaitant louer une Dixel dans Excel et aussi comment ouvrir Outlook avec un mail préparé à envoyer à ces clients.

Pour pouvoir appeler les composants COM, nous devons au préalable rajouter une référence à Microsoft.CSharp dans notre application Silverlight. En effet, l'appel à ces composants utilise les types dynamiques introduits avec C# 4. La gestion de COM depuis Silverlight se base sur la classe AutomationFactory.

```
if (!AutomationFactory.IsAvailable)
    return;
dynamic excel = AutomationFactory.CreateObject(«Excel.Application»);
```

Premièrement, on vérifie que l'on peut instancier des composants grâce à la propriété statique IsAvailable de la classe AutomationFactory. Ensuite on instancie le composant avec la méthode CreateObject qui prend en paramètre le nom du composant, ici Excel, et on crée un nouveau document Excel en récupérant la feuille active.

```
dynamic workbook = excel.workbooks;
workbook.Add();

dynamic sheet = excel.ActiveSheet;
```

Il ne reste qu'à remplir les cellules de la feuille et de rendre visible Excel.

```
dynamic cell = null;
int i = 1;

foreach (var item in customers)
{
    cell = sheet.Cells[i, 1];
    cell.Value = item.Name;
    cell.ColumnWidth = 100;
    ...
    i++;
}

excel.Visible = true;
```

Dans la page des détails d'un client on a un bouton qui nous permet d'ouvrir Outlook pour envoyer un email. La procédure pour utiliser le composant COM d'Outlook est sensiblement la même que celle pour Excel. Voici comment on procède :

```
dynamic outlook = AutomationFactory.CreateObject(«Outlook.
Application»);
dynamic mail = outlook.CreateItem(0);
mail.To = _customer.Email;
mail.Subject = «About your Dixel»;
```

```
mail.Body = «We have a Dixel for you !»;
mail.Display();
```

ACCÈS AU SYSTÈME DE FICHIERS

Précédemment nous avons ajouté la possibilité d'ouvrir Excel, nous allons maintenant permettre à l'utilisateur de sauvegarder directement la liste des clients souhaitant louer une Dixel dans un fichier Excel sur le disque dur de l'ordinateur sans passer par l'affichage de ce dernier et sans utiliser de SaveFileDialog.

L'accès aux fichiers et aux dossiers est limité dans Silverlight 4 aux dossiers Mes Documents, Mes Images, Ma Musique et Mes Vidéos. Attention à ne pas confondre ces dossiers avec les listes Documents, Musique etc... présentes dans Windows 7. Dans le cadre de notre application d'exemple le fichier Excel sera sauvegardé dans le dossier Mes Documents\ContosoDashboard.

Dans un premier temps, nous devons récupérer le chemin vers le dossier Mes Documents. Cette opération se fait obligatoirement en utilisant la méthode GetFolderPath de la classe Environment :

```
string myDocuments =
Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);
string saveDirectory = string.Format(«{0}\ContosoDashboard»,
myDocuments);
```

On vérifie si le dossier existe déjà sinon on le crée :

```
if (!Directory.Exists(saveDirectory))
    Directory.CreateDirectory(saveDirectory);
```

On vérifie si le fichier Excel existe déjà, si oui on le supprime :

```
string path = string.Format(«{0}\{1}», saveDirectory, «customers.
xlsx»);

if (File.Exists(path))
    File.Delete(path);
```

On initialise la feuille Excel :

```
dynamic excel = AutomationFactory.CreateObject(«Excel.Application»);
dynamic excelWorkBook = excel.Workbooks.Add();
dynamic sheet = excelWorkBook.ActiveSheet();
```

A partir de ce moment la mécanique est la même que pour afficher directement la feuille Excel. Seule différence notable, la sauvegarde de la feuille et la fermeture d'Excel :

```
excelWorkBook.SaveAs(path);
excelWorkBook.Close();
excel.Quit();
```

Le fichier est désormais créé dans le dossier et sans intervention de l'utilisateur.

MODE PLEIN ÉCRAN

Une application Silverlight a la possibilité de passer en plein écran suite à une interaction de l'utilisateur tel qu'un clic sur un bouton. Le



plugin Silverlight demande alors à l'utilisateur s'il accepte que l'application passe en plein écran et s'il accepte, un message l'informe qu'il peut sortir du mode plein écran en appuyant sur la touche Echap. De plus, dans ce mode de fonctionnement, le plugin va intercepter la plupart des événements clavier rendant impossible la saisie de données dans des champs textes. Le mode plein écran de Silverlight classique est alors limité à de la consultation de données ou à la lecture de musiques ou de vidéos par exemple. En mode de confiance élevée, l'application dispose des droits nécessaires pour passer en mode plein écran sans demander le consentement de l'utilisateur. De plus, le plugin n'intercepte plus les événements clavier, ce qui induit que la saisie de texte est de nouveau possible et que l'utilisateur ne peut plus sortir du mode plein écran juste en appuyant sur la touche Echap. C'est au développeur de fournir un mécanisme pour permettre à l'utilisateur de revenir en mode fenêtre, si toutefois il lui en laisse la possibilité.

Voici le code de gestionnaire de l'évènement clic d'un bouton qui active et désactive alternativement le mode plein écran :

```
Application.Current.Host.Content.IsFullScreen ^= true;
```

MODE HORS LIGNE

Une application OOB peut s'exécuter alors que tout accès réseau est coupé. C'est alors à la charge du développeur de fournir une expérience d'utilisation de l'application en mode déconnecté s'il le souhaite. Pour savoir si l'application est en ligne on utilise la méthode suivante :

```
NetworkInterface.GetIsNetworkAvailable()
```

Pour savoir lorsque l'état du réseau change on doit procéder comme suit :

```
NetworkChange.NetworkAddressChanged += NetworkChange_NetworkAddressChanged;
```

En combinant ces deux méthodes on peut surveiller l'état du réseau en permanence. Attention toutefois, la méthode `GetIsNetworkAvailable` indique si votre ordinateur dispose d'une carte réseau active et connectée. Si vous êtes dans le train sans réseau et que vous tentez de vous connecter sur un service hébergé sur votre poste local, cette méthode renverra faux. Alors gardez cela à l'esprit lorsque vous débutez hors-ligne.

L'implémentation d'un mode déconnecté se fait généralement en téléchargeant les données nécessaires au fonctionnement de l'application tant que celle-ci est connectée au réseau et en se resynchronisant avec le serveur lorsque l'application revient en ligne. La zone où l'on va stocker les informations chargées est l'isolated storage. Dans une application Silverlight classique l'isolated storage a une taille de 1Mo par défaut, mais une application hors du navigateur dispose par défaut de 20Mo. Cela permet de stocker plus d'informations sans avoir besoin de demander une augmentation de quota à l'utilisateur. L'utilisation de l'isolated storage n'a ici rien de particulier par rapport à du Silverlight classique. L'absence de base de données intégrée dans Silverlight impose de sérialiser les entités que nous souhaitons sauvegarder dans l'isolated storage pour les persister. C'est un travail fastidieux mais nécessaire. Certaines bibliothèques tierces nous permettent cependant de grandement nous simplifier la vie pour gérer cette sérialisation de manière efficace et

plus proche d'un modèle de base de données. Parmi ces bibliothèques il en existe une développée par Jesse Liberty et du nom de Sterling (<http://sterling.codeplex.com/>).

NATIVE EXTENSIONS FOR MICROSOFT SILVERLIGHT

Basée sur le support de COM Automation par Silverlight 4, cette bibliothèque de composants COM englobe des fonctionnalités natives du système à destination de Silverlight OOB. Elle est disponible à l'adresse suivante :

<http://archive.msdn.microsoft.com/nsl>

Parmi les fonctionnalités exposées se trouvent la gestion avancée du tactile, le support de la barre des tâches de Windows 7, la possibilité de bloquer le nombre d'instances d'une application Silverlight OOB à une, la gestion des capteurs et du Speech, le support de l'accès aux périphériques externes tels que les appareils photos, les téléphones les baladeurs, etc.

QU'ATTENDRE DE SILVERLIGHT 5 ?

Silverlight 5 apporte tout un lot de nouveautés qui concernent son mode hors du navigateur avec un niveau de confiance élevée. Il sera possible d'ouvrir de multiples fenêtres, cependant celles-ci ne seront pas modales. On aura aussi accès à tout le système de fichier de l'ordinateur mais en utilisant les droits de l'utilisateur courant. On pourra aussi faire appel à du code non managé, ce qui ouvre tout un champ de possibilités à explorer.

CONCLUSION

Grâce à Silverlight OOB nous avons la possibilité de développer des applications bureau riches et fonctionnant à la fois sur des systèmes Windows et Mac OS. De plus, le plugin Silverlight et les applications OOB en elles-mêmes peuvent être déployées sur tous les postes d'une entreprise grâce à des stratégies de groupe sans que l'utilisateur du poste ne s'en préoccupe. Elles peuvent fonctionner en mode déconnecté pour une plus grande flexibilité et disposent à cet effet d'une taille d'isolated storage plus importante par défaut. On a la possibilité d'intégrer un navigateur au sein de l'application ou d'afficher des notifications toast. Les applications peuvent aussi se mettre à jour automatiquement sans intervention humaine.

En mode de confiance élevée on bénéficie d'une grande intégration au système, notamment grâce à COM Automation et aux fenêtres personnalisées et à une gestion du réseau comportant moins de contraintes de sécurité. On peut accéder aux dossiers Mes Documents, Mes Images etc. et passer l'application en mode plein écran. Avec Silverlight 5 on pourra en plus accéder à tout le système de fichier, au presse-papier et ouvrir d'autres fenêtres.

Toutes ces fonctionnalités font de Silverlight OOB une plateforme de choix pour le développement d'applications bureau riches avec laquelle des projets conséquents peuvent être développés à la fois pour des utilisations internes ou à destination du grand public.

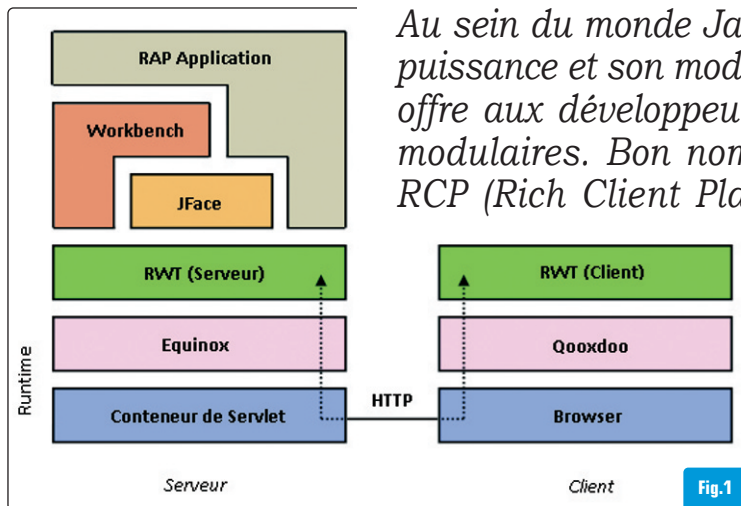
Vous pouvez retrouver les sources de l'application de démonstration sur codeplex (<http://sloobcontoso.codeplex.com/>) afin de pouvoir tester vous-mêmes la plupart des fonctionnalités évoquées ci-dessus.

John Thiriet

Expert Silverlight MCNEXT

<http://blog.john-thiriet.com/>

Rich Ajax Platform : le meilleur d'Eclipse pour le Web



Au sein du monde Java, la plateforme Eclipse est réputée pour sa puissance et son modèle de programmation orienté composants qui offre aux développeurs la possibilité de réaliser des applications modulaires. Bon nombre d'entreprises ont ainsi adopté Eclipse RCP (Rich Client Platform) pour la réalisation de leurs clients riches. La fondation Eclipse a décidé d'amener RCP vers le Web. Le projet Eclipse RAP (Rich Ajax Platform) était né !

mentation OSGI de la fondation Eclipse. Côté client, RWT utilise le framework Javascript open-source Qooxdoo afin d'implémenter les widgets riches définis via l'API Java. Le rendu est donc bien transparent pour le développeur mais il reste possible pour des besoins de widgets complexes de créer ses propres widgets spécifiques. Bien entendu, dans ce cas-là il sera nécessaire de réaliser le développement Javascript correspondant pour gérer la partie client de ces widgets. La communication entre client et serveur est réalisée via des appels AJAX. A la différence de GWT qui essaie de réduire au maximum les appels serveurs pour le traitement des événements IHM en produisant le Javascript nécessaire, RAP adopte une approche centrée sur le serveur au sein de laquelle chaque action utilisateur va entraîner un appel AJAX permettant l'exécution du traitement correspondant côté serveur et le renvoi du nouveau rendu de la page côté client. Ainsi, le simple fait de déplier le nœud d'un arbre de données provoquera un appel serveur par exemple.

ARCHITECTURE RAP

RAP tire tout naturellement son architecture de celle d'Eclipse RCP [Fig.1]. Ainsi, les couches hautes sont les mêmes pour les deux technologies. Toutes deux sont basées sur le Workbench Eclipse qui permet de manipuler fenêtres, perspectives, vues et éditeurs au sein de l'application.

La petite spécificité au niveau du Workbench utilisé par RAP vient de la prise en compte de l'environnement multi-utilisateurs puisqu'on est sur le Web et d'une gestion plus fine des contraintes mémoire qui en découlent. Au niveau de l'interface graphique, l'API de haut niveau JFace donne au développeur l'accès à un ensemble de composants haut niveau fournissant notamment des arbres, des tables de données ou bien des boîtes de dialogue. Au niveau du toolkit graphique, RAP utilise une implémentation de SWT spécifique générant un rendu HTML. De fait, le développeur réalise son IHM de la même manière qu'il le ferait pour une application RCP desktop et c'est ensuite au runtime que la différence est faite suivant la cible retenue pour l'application. Nommée RWT (RAP Widget Toolkit), cette implémentation va diviser une requête HTTP reçue en différentes phases qui seront exécutées séquentiellement. Celles-ci sont au nombre de 4 :

- Prepare UI Root, en charge de l'invocation des points d'entrée de l'interface graphique de l'application.
- Read Data, qui va récupérer les paramètres envoyés lors de la requête par le client.
- Process Action, phase durant laquelle les traitements liés à des actions utilisateur vont être exécutés.
- Render, phase au cours de laquelle le code HTML et le code Javascript de l'application sera créé et envoyé au navigateur.

La connaissance approfondie de ces phases n'est pas nécessaire pour développer des applications RAP mais il est intéressant de remarquer que le cycle de vie retenu par les créateurs du framework se rapproche de ce qui est proposé par JSF notamment. RAP étant basé sur la plateforme Eclipse, une application RAP est un bundle OSGI qui tourne au sein du conteneur Equinox qui est l'implé-

MODÈLE DE PROGRAMMATION

Le modèle de programmation de RAP est basé sur le Workbench Eclipse, composant très puissant qui est à la base de la plateforme Eclipse. Réaliser l'interface graphique d'une application RAP va ainsi consister à s'appuyer sur le Workbench au sein duquel on ajoutera les différents éléments composant l'IHM. Les différentes notions associées à ce dernier sont bien connues des utilisateurs de l'IDE Eclipse et sont présentées dans le diagramme de classes de la [Fig.2].

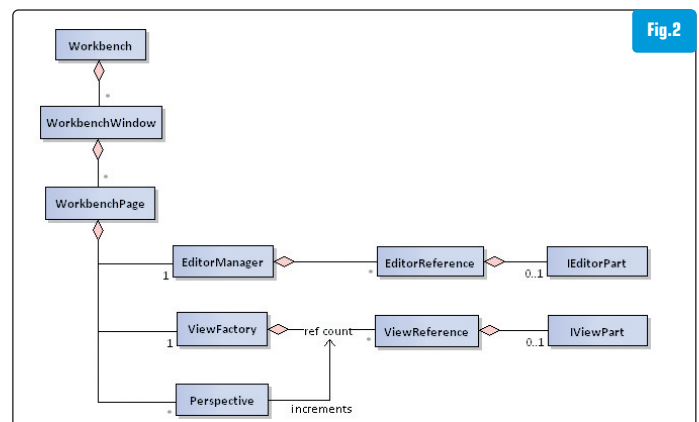


Diagramme de classes simplifié du Workbench

Conceptuellement, le Workbench représente un ensemble de fenêtres qui peuvent contenir des menus, des barres d'outils et plusieurs perspectives le cas échéant. Ces différentes fenêtres sont modélisées via la classe `WorkbenchWindow` qui peut contenir plusieurs éléments de type `WorkbenchPage` représentant une configuration donnée de perspectives, de vues et d'éditeurs. Un projet RAP étant un bundle OSGI, le lancement d'une application est réalisé par le conteneur OSGI Equinox. A son lancement, l'instanciation du Workbench est réalisée via la méthode `createUI` du point d'entrée de l'application qui est une classe implémentant l'interface `IEntryPoint`. La configuration du Workbench est réalisée durant ce lancement via une classe dérivant de `WorkbenchAdvisor` au sein de laquelle il est nécessaire de définir l'identifiant de la perspective par défaut qui sera ajoutée au Workbench en cours de création, mais également le configurateur de fenêtres utilisé. Ce dernier est représenté par une classe héritant de `WorkbenchWindowAdvisor` et permet de définir les menus ainsi que les barres d'outils qui seront ajoutés au Workbench.

STRUCTURE D'UNE APPLICATION

Parfaitement intégré à la plateforme Eclipse, RAP bénéficie de facto de l'ensemble de l'outillage proposé par l'IDE pour la création d'applications. Cela va des assistants de création de plugins à l'éditeur de fichier de configuration en passant par le projet `WindowBuilder` qui bien qu'encore en incubation permet de construire les interfaces graphiques des applications RAP. Une fois le package Eclipse dédié aux plateformes RAP et RCP téléchargé et la configuration du SDK RAP réalisée, la création d'un projet RAP se réalise simplement en créant un nouveau projet de type plugin dans le wizard prévu à cet effet. Durant le processus de création de l'application, on choisira Eclipse comme plateforme cible et nous n'utiliserons pas d'Activator spécifique pour le bundle. Enfin, il est intéressant de démarrer en

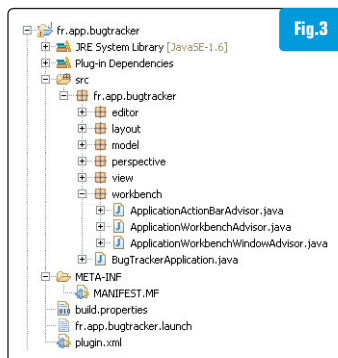


Fig.3

Arborescence d'une application RAP

utilisant un template d'application RAP afin de comprendre plus rapidement la structure d'une application. L'arborescence du projet, une fois ces étapes suivies, est présentée à la figure 3.

On constate qu'une application RAP respecte bien l'arborescence d'un projet de plugin Eclipse avec la présence du fichier de configuration `plugin.xml` et de celle du fichier `MANIFEST.MF` au sein du dossier `META-INF`. Ce dernier décrit le plugin comme un bundle OSGI et spécifie ses caractéristiques. L'environnement d'exécution Java retenu ainsi que les dépendances du plugin sont également détaillées. Enfin, le dossier `src` contient de façon classique les sources utilisées pour construire l'application.

On constate qu'une application RAP respecte bien l'arborescence d'un projet de plugin Eclipse avec la présence du fichier de configuration `plugin.xml` et de celle du fichier `MANIFEST.MF` au sein du dossier `META-INF`. Ce dernier décrit le plugin comme un bundle OSGI et spécifie ses caractéristiques. L'environnement d'exécution Java retenu ainsi que les dépendances du plugin sont également détaillées. Enfin, le dossier `src` contient de façon classique les sources utilisées pour construire l'application.

CRÉATION D'UNE APPLICATION

Nommée `fr.app.bugtracker`, notre application RAP possède un point d'entrée que nous créons dans la classe `BugTrackerApplication`. Son contenu est assez simple puisque seule la méthode `createUI` y est présente. C'est au sein de celle-ci que l'instanciation du Workbench est faite :

```
public class BugTrackerApplication implements IEntryPoint {
    public int createUI() {
        // récupération du display
        Display display = PlatformUI.createDisplay();
        // création du configurateur de workbench
        WorkbenchAdvisor advisor = new ApplicationWorkbenchAdvisor();
        // création et lancement du workbench
        return PlatformUI.createAndRunWorkbench(display, advisor);
    }
}
```

Le configurateur de Workbench pour notre application est implémenté grâce à la classe `ApplicationWorkbenchAdvisor` au sein de laquelle nous définissons le configurateur de fenêtres pour le Workbench ainsi que la perspective par défaut de ce dernier en la référant via un identifiant unique :

```
public class ApplicationWorkbenchAdvisor extends WorkbenchAdvisor {

    public WorkbenchWindowAdvisor createWorkbenchWindowAdvisor (IWorkbenchWindowConfigurer configurateur) {
        return new ApplicationWorkbenchWindowAdvisor(configurateur);
    }

    public String getInitialWindowPerspectiveId() {
        return Perspective.PERSPECTIVE_ID;
    }
}
```

Le travail de configuration des fenêtres du Workbench est délégué à une autre classe nommée `ApplicationWorkbenchWindowAdvisor` :

```
public class ApplicationWorkbenchWindowAdvisor extends WorkbenchWindowAdvisor {

    public ApplicationWorkbenchWindowAdvisor(IWorkbenchWindowConfigurer configurateur) {
        super(configurateur);
    }

    public ActionBarAdvisor createActionBarAdvisor(IActionBarConfigurer configurateur) {
        return new ApplicationActionBarAdvisor(configurateur);
    }

    public void preWindowOpen() {
        IWorkbenchWindowConfigurer configurateur = getWindowConfigurer();
        configurateur.setShowMenuBar(true);
        configurateur.setShowCoolBar(true);
        configurateur.setShowPerspectiveBar(false);
        configurateur.setShowProgressIndicator(true);
        // Affichage de l'application sur toute la zone d'affichage du navigateur
        getWindowConfigurer().setShellStyle(SWT.NO_TRIM);
        configurateur.setTitle("BugTracker Application");
    }
}
```

La méthode `preWindowOpen` nous permet de configurer le comportement d'une fenêtre de Workbench avant son ouverture en spécifiant notamment que l'on souhaite l'afficher sur l'ensemble de la zone d'affichage du navigateur. Enfin, la création du menu ou de la barre d'outils de l'application est réalisée au sein de la classe `ApplicationActionBarAdvisor`. On le voit, la séparation des responsabilités au sein du Workbench est bien respectée conférant ainsi à ce dernier une grande modularité.

POINTS D'EXTENSION

Autre notion essentielle au cœur des applications RAP, les extensions et les points d'extension qui confèrent à Eclipse sa modularité en permettant d'avoir un couplage faible entre composants et bundles. Concrètement, un point d'extension est ce qui va permettre à un plugin de la plateforme Eclipse d'autoriser d'autres plugins à l'utiliser en lui ajoutant des fonctionnalités supplémentaires en se basant sur le contrat défini par le point d'extension. L'extension va quant à elle permettre à un plugin d'utiliser des points d'extension afin d'étendre ses fonctionnalités. La gestion des points d'extension et des extensions est réalisée par le conteneur OSGi Equinox au run-time et se base sur le fichier de configuration de l'application RAP. La construction de l'interface graphique de cette dernière sera de fait réalisée en étendant le Workbench grâce à des extensions pouvant être des vues, des éditeurs ou des perspectives. Afin de définir le point d'entrée de notre application et sa perspective par défaut, nous allons utiliser ce mécanisme. Pour ce faire, il faut ouvrir le fichier `MANIFEST.MF` afin de configurer le bundle représenté par notre application RAP. Une fois le fichier ouvert, Eclipse propose un éditeur qui va nous assister durant la définition des extensions utilisées [Fig.4]. Le point d'entrée de l'application étend le point d'extension `org.eclipse.rap.ui.entrypoint`. Nous lui définissons un identifiant unique ainsi que la classe de notre application qui étend effectivement ses fonctionnalités et son paramètre d'entrée à savoir la valeur `home`. Enfin, nous définissons la perspective par défaut du Workbench de notre application RAP en utilisant le point d'extension `org.eclipse.ui.perspectives` dont l'identifiant correspond à celui référencé dans la classe `ApplicationWorkbenchAdvisor`.

VUES ET EDITEURS

Le socle de l'interface graphique de notre application RAP mis en place, il est temps de passer à la réalisation de sa vue principale. Cette vue aura pour rôle d'afficher une liste de bugs classés par statuts. Elle vient enrichir le Workbench de notre application ce qui impose de la déclarer comme extension au sein de son fichier de configuration en lui affectant

un identifiant unique. Elle étend la classe `ViewPart` et utilise l'API `JFace` pour manier un arbre de données. Le contenu de la vue est créé au sein de la méthode surchargée `createPartControl` qui est invoquée par le Workbench lorsque la vue est affichée :

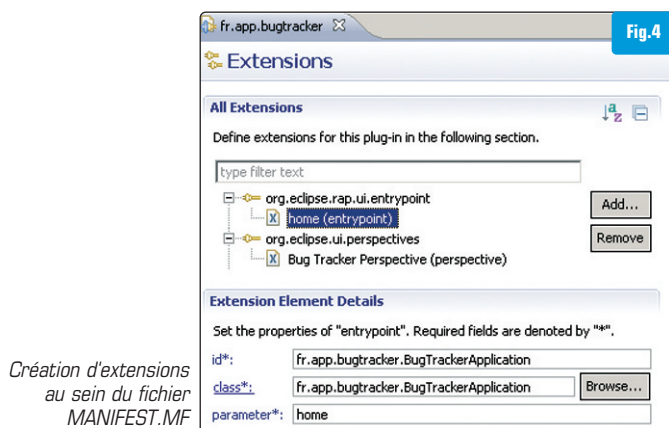
```
public void createPartControl(Composite parent) {
    viewer = new TreeViewer(parent, SWT.MULTI | SWT.H_SCROLL |
        SWT.V_SCROLL | SWT.BORDER);
    // on configure l'arbre de données
    viewer.setContentProvider(new ViewContentProvider());
    viewer.setLabelProvider(new ViewLabelProvider());
    viewer.setInput(createModel());
    viewer.expandAll();
    // ajout d'un listener appelé lors du double click d'un élément
    // de l'arbre
    viewer.addDoubleClickListener(new IDoubleClickListener() {
        @Override
        public void doubleClick(DoubleClickEvent event) {
            ISelection selection = viewer.getSelection();

            if (selection instanceof IStructuredSelection) {
                Object element = ((IStructuredSelection)selection).getFirstElement();

                if (element instanceof TreeObject) {
                    // on travaille seulement sur les tree object ici
                    TreeObject treeObject = (TreeObject) element;
                    // on récupère le bug associé à la sélection
                    Bug bug = treeObject.getBug();
                    // creation des données d'entrée pour l'éditeur de bug
                    BugEditorInput input = new BugEditorInput(bug);

                    try {
                        // on ouvre l'éditeur dans la partie du Workbench prévue
                        // à cet effet
                        getSite().getPage().openEditor(input, BugEditor.ID);
                    } catch (PartInitException e) {
                        throw new RuntimeException(BugEditor.ID + " not found");
                    }
                }
            }
        }
    });
}
```

Le double-click sur un élément de l'arbre de données provoque l'ouverture d'un éditeur au sein du Workbench de notre application RAP. Cet éditeur permettant d'éditer les informations concernant le bug sélectionné. Il étend le Workbench et son comportement est défini au sein de la classe `BugEditor` qui hérite de `EditorPart`. Comme précédemment pour la vue, elle surcharge la méthode `createPartControl` au sein de laquelle son contenu visuel est créé en s'appuyant sur les composants bas niveaux proposés par l'API `SWT`. En outre, la méthode `init` qui est également surchargée est appelée par le Workbench au moment de l'initialisation de l'éditeur et va permettre de récupérer le bug à éditer via l'objet `BugEditorInput`. Enfin, il est nécessaire d'ajouter la vue de base de notre application RAP au sein de la perspective par défaut de notre Workbench comme ceci :




```
public class Perspective implements IPerspectiveFactory {
    public static final String PERSPECTIVE_ID = "fr.app.bugtracker.
perspective";

    public void createInitialLayout(IPageLayout layout) {
        String editorArea = layout.getEditorArea();
        layout.setEditorAreaVisible(false);
        layout.addStandaloneView(BugListView.ID, true, IPageLayout.
LEFT, 0.25f, editorArea);
        layout.getViewLayout(BugListView.ID).setCloseable(false);
    }
}
```

L'exécution de notre application RAP nous permet d'avoir un premier aperçu de son aspect [Fig.5]. On remarque que la servlet par défaut est nommée rap et que la variable passée en paramètre correspond au nom du point d'entrée utilisé pour l'application.

THÈMES

Bien que notre application soit parfaitement fonctionnelle, son aspect visuel, très proche de ce qu'on peut voir avec Eclipse, laisse clairement à désirer pour une application web RIA. Pour adresser cette problématique de customisation, le framework RAP propose un mécanisme de thèmes via l'extension org.eclipse.rap.ui.themes qui prend en entrée un fichier CSS et va appliquer les propriétés de style définies sur les composants de l'interface graphique. Les sélecteurs utilisés par RAP sont quelque peu différents de ce qui se fait habituellement puisqu'ici on utilisera le type d'un composant pour le sélectionner et lui affecter des propriétés de style. Ainsi, pour spécifier la couleur du label d'un bouton, on utilisera le code suivant :

```
Button {color : rgb(255, 0, 0); }
```

Outre ce mécanisme de thèmes, le framework RAP offre la possibilité de définir un aspect visuel commun à l'ensemble des applications d'une entreprise via le branding. Ce mécanisme se présente sous la

forme d'une extension org.eclipse.rap.ui.branding qui s'appuie sur la définition d'un thème et d'un look and feel qui seront appliqués à l'application l'utilisant. En sus, elle permet de définir le nom de la servlet par défaut de l'application. Afin d'améliorer le rendu de notre applica-

tion de gestion des bugs, nous appliquons le thème business proposé en standard par RAP à notre branding. Le résultat est impressionnant et l'on obtient une application web client riche parfaitement opérationnelle avec un rendu visuel professionnel [Fig.6]. La copie d'écran du rendu final de l'application RAP permet de voir que la barre d'outils est affichée sous forme d'une cool bar suivant la dénomination RAP et qu'un logo spécifique a été affiché.

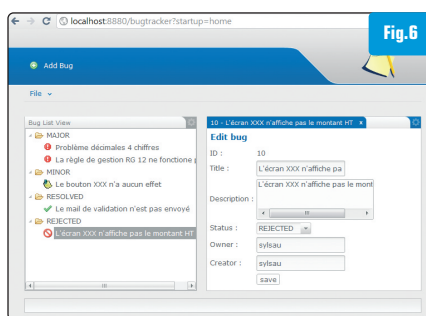
DÉPLOIEMENT

Le déploiement d'une application RAP peut être réalisé suivant 2 scénarios. Le premier consiste à faire tourner l'application au sein du conteneur de servlets Jetty exécuté en tant que bundle OSGI au sein d'Equinox. Ce mode de déploiement, utilisé pour tester l'application au sein de l'IDE Eclipse, est celui recommandé car plus léger et utilisant pleinement la puissance d'OSGI. Cependant, il est fort probable qu'au sein d'une entreprise il ne soit pas possible de mettre en place une infrastructure basée sur OSGI et que certaines contraintes imposent de déployer l'application RAP sous forme de war au sein du serveur web en place. Nous allons donc nous intéresser ici à cette seconde méthode. Moins naturelle, elle était réellement complexe jusqu'alors, mais les développeurs de la plateforme Eclipse ont pris conscience de l'intérêt de ce mode de déploiement et ont donc décidé d'investir en ce sens. De fait, la nouvelle mouture de l'IDE Eclipse nommée Indigo bénéficie d'un plugin spécifique permettant la création d'un war à partir d'un projet RAP. Encore en incubation, il est disponible à l'adresse suivante : <http://download.eclipse.org/eclipse/downloads/eclipse-releases/3.7/>. Une fois le plugin installé, il est nécessaire de créer une nouvelle configuration de type WAR Product pour notre application en précisant que celle-ci est basée sur la configuration de lancement de l'application au sein d'Eclipse. L'assistant de création va se charger de créer le dossier WEB-INF au sein de notre projet RAP ainsi que le fichier web.xml qui permettra le lancement de l'application web au sein du conteneur de servlets. La configuration créée, un export du projet en tant que war est désormais possible, ce qui rend l'application RAP déployable au sein de n'importe quel conteneur de servlets du marché. Cette solution marche parfaitement mais le poids de l'application produite s'avère plus important. En effet, l'application RAP doit embarquer en son sein un certain nombre de dépendances permettant à Equinox de tourner lorsque l'application est lancée. La servlet bridge du moteur OSGI Equinox se chargeant de faire le lien entre les appels HTTP reçus et le bundle de notre application RAP qui tourne en son sein.

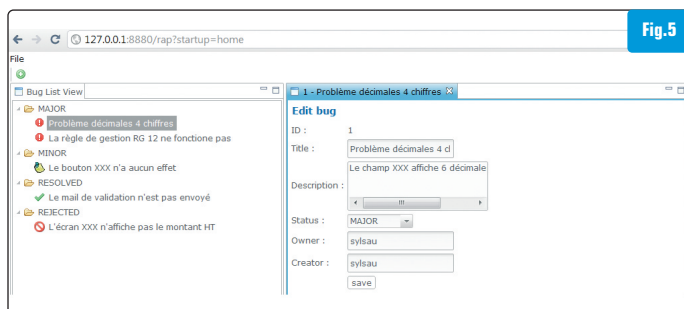
CONCLUSION

Le framework RAP tire parti de la puissance de la plateforme Eclipse et de son outillage pour faciliter la création d'applications webs RIA. Pendant de RCP sur le Web, il permet à des entreprises ayant investi dans la plateforme Eclipse de pouvoir réutiliser leurs applications RCP en les portant sur le Web à moindre coût tout en unifiant le développement. En somme, un seul développement permet d'atteindre à la fois les cibles desktop et web avec un niveau de fonctionnalités quasi similaire. Centré sur le serveur de par son architecture, RAP réalise énormément d'appels AJAX, ce qui nuit malheureusement à la réactivité des applications et vient lui imposer certaines limites techniques.

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



Rendu final de notre application RAP



Screenshot de notre application

Rejouer les actions des utilisateurs sous Windows

Sous Windows, les actions de l'utilisateur, clavier, souris, etc. sont converties en des messages placés dans des files. Un mécanisme très intéressant, les « hook », permet d'intervenir sur les files de messages.

Pour qu'un système multitâche puisse fonctionner, il doit traiter les événements de manière asynchrone. En effet, ces événements survenant à des moments imprévisibles, il n'est pas question d'attendre qu'ils se produisent sous peine de bloquer le système entier. Les événements susceptibles d'être gérés par le système sont très nombreux. Les actions physiques de l'utilisateur sur le clavier et la souris sont les plus évidents, et ce sont eux qui nous intéressent aujourd'hui. Schématiquement, Windows résout le problème de ces événements en les plaçant à la queue leu leu dans une file. Un thread du système dispatche ensuite chaque événement relatif à une application dans une autre file, dédiée à l'application concernée. L'application, dans son thread principal, à charge de "pomper" les événements, c'est-à-dire de les retirer de sa file, et d'y réagir de la façon appropriée dans sa procédure de fenêtre. Il est possible de suivre tout ce trafic de messages et même d'intervenir dessus, au moyen des hooks. Les hooks sont de petites fonctions que l'on installe par programmation, et qui sont appelées par le système qui leur passe les messages de la file qui les concerne. Travailler avec les hooks est amusant, et permet de donner libre cours à son imagination, pour le meilleur et pour le pire. C'est ainsi que les keyloggers, programmes généralement malveillants, utilisent le principe des hooks pour enregistrer les saisies claviers d'un utilisateur. Nous allons, quant à nous, jouer avec les messages de la souris dans cet article, notamment pour en rejouer les déplacements après coup. Mais si vous souhaitez coder un keylogger (pour une raison louable bien sûr !), nos exemples seront facilement adaptables. Nos exemples ont été écrits en C++ avec Visual Studio 2008. Vous les adapterez facilement à votre compilateur préféré. Le seul point un petit peu difficile est, éventuellement, l'édition de liens dynamique avec les bibliothèques dll, ainsi que nous le verrons plus loin. Pour cet article, nous supposons que le lecteur dispose déjà de solides bases en programmation Win32.

1 HOOKUMENIQUE

Le champ d'action des hooks est assez universel dans Windows. Outre les événements relatifs aux actions de l'utilisateur, il est possible d'en poser pour savoir si le système rentre dans un état inactif (Idle), Pour monitorer les notifications dans le Shell graphique, ou les écritures dans les journaux systèmes, etc. Il est même possible de placer un hook pour déboguer les autres hooks. Dans tous les cas, un hook est une fonction prototypée ainsi :

```
LRESULT CALLBACK FonctionHook(int nCode, WORD wParam, DWORD lParam);
```

Les arguments wParam et lParam sont, sans surprise, les arguments des messages interceptés. Une attention toute particulière doit être accordée à l'argument nCode ainsi qu'à la valeur retournée

par la fonction. Si l'argument nCode a une valeur négative, il est interdit d'agir sur le message intercepté par le hook. Après nous être informés de la teneur du message, nous devons laisser celui-ci suivre son cours soit en retournant false, soit en retournant la valeur donnée par l'API CallNextHookEx. Dans le premier cas, le message va suivre son cours dans le système, mais sans être vu par d'autres hooks éventuels. En effet de nombreux hooks peuvent être mis en place pour un même type de message. Ils sont alors chaînés. Le dernier installé étant le premier de la chaîne et donc le premier servi. Priver d'autres éventuels hooks d'informations n'est pas forcément bien élevé en programmation saine. Mais cela peut aussi être le moyen de désactiver un éventuel keylogger :) Dans le second cas, le message est passé au hook suivant (s'il existe) dans la chaîne et ainsi de suite. Si nCode a une valeur positive ou nulle, il est possible de faire comme précédemment. Mais il est aussi possible de passer le message à la trappe en retournant true. Dans ce cas, non seulement les autres hooks ne le verront pas passer, mais l'application qui devait le recevoir ne le verra pas non plus :) Enfin il est nécessaire de savoir qu'il existe deux niveaux de hook: application et système. Un hook de niveau application ne verra passer que les messages destinés à cette application, ceci, juste avant qu'elle ne les reçoive. Dans ce cas, la procédure de hook peut résider dans le code de l'application sans problème, car elle sera appelée par le thread principal de l'application. Un hook de niveau système voit passer les messages dans la file principale, avant le dispatching de ceux-ci dans les files dédiées aux applications. Le code d'un tel hook ne peut résider dans une application. Il doit au contraire résider dans une bibliothèque partagée (dll) car il est susceptible d'être appelé par n'importe quel thread.

2 POUR SE FAIRE LA MAIN

Tous nos exemples sont des applications Win32 pures et dures, donc par définition d'un code assez verbeux. C'est pourquoi nous n'en donnons que les extraits importants ici, mais vous pouvez les télécharger au complet sur notre site, à la page dédiée au sommaire de ce numéro. Notre premier exemple, HookMouseInWindows, est une application qui trace constamment dans sa fenêtre les mouvements de la souris au-dessus de cette même fenêtre. Lorsque l'utilisateur clique sur le menu "Go!" un hook est installé pour une durée de trois secondes. Ce hook passe à la trappe les messages de la souris. Ainsi, l'application ne suit plus ces mouvements pendant ce temps. Voici le code du hook :

```
HHOOK HookHandle; // variable globale
```

```
LRESULT CALLBACK MaMouseProc(int nCode, WPARAM wParam, LPARAM lParam)
{
```



```
if(nCode < 0)
{
    return CallNextHookEx(HookHandle, nCode, wParam, lParam);
}
else
{
    return true;
}
}
```

Comme expliqué précédemment, si la valeur de nCode nous y autorise, nous retournons true, ce qui est nécessaire et suffisant pour tuer les messages. Voici la partie intéressante du code de la procédure de fenêtre qui installe et supprime le hook, et qui réagit aux événements significatifs pour notre exemple :

```
static UINT_PTR timer = 0;
BOOL unhooked;
static POINT

case ID_FILE_GO:
    // Installer le hook des événements souris
    HookHandle = SetWindowsHookEx(WH_MOUSE, // Type de hook
        MaMouseProc,
        hInst, // Handle d'instance de notre programme.
        GetCurrentThreadId() // Identifiant du thread courant,
            // i.e celui de la pompe
            // à message de l'application
    );
    if(HookHandle == NULL)
    {
        MessageBox(hWnd, L"Oups hook", L"Oups!", MB_OK);
        PostQuitMessage(0);
        break;
    }
    // Démarrer le timer
    timer = SetTimer(hWnd, 1000, 3000, NULL);
    if(timer == 0)
    {
        MessageBox(hWnd, L"Oups timer", L"Oups!", MB_OK);
        PostQuitMessage(0);
    }
    break;
case WM_TIMER:
    // Arrêter le timer
    KillTimer(hWnd, timer);
    // Enlever le hook
    unhooked = UnhookWindowsHookEx(HookHandle);
    if(!unhooked)
    {
        MessageBox(hWnd, L"Oups Unhook", L"Oups!", MB_OK);
        PostQuitMessage(0);
    }
    break;
case WM_MOUSEMOVE:
    hdc = GetDC(hWnd);
    MoveToEx(hdc, pt.x, pt.y, NULL);
    LineTo(hdc, pt.x = LOWORD(lParam),
```

```
pt.y = HIWORD(lParam));
ReleaseDC(hWnd, hdc);
break;
```

Observons bien le fonctionnement de notre exemple. Notre hook ne fait pas dans la dentelle puisqu'il tue TOUS les messages relatifs à la souris : déplacement, bouton enfoncé, etc. Cependant quand vous déplacez votre souris, vous continuez à voir le pointeur bouger. Pourquoi ? Parce que notre hook est local. Son code est intégré à notre application. En tant que hook local, il n'agit que sur les événements à destination de l'application. Or, le pointeur de la souris que vous voyez se déplacer appartient au bureau de Windows, qui est une autre application.

3 REJOUER DES ÉVÉNEMENTS

L'exemple ReplayMouseInWindows montre comment enregistrer les événements que notre hook passe à la trappe, pour les rejouer ensuite, après l'expiration du timer de 3 secondes. En outre, cette fois nous affinons le code de notre hook, afin que celui-ci ne traite que les événements relatifs aux déplacements de la souris, et non, par exemple, les clics :

```
#include <vector>
using namespace std;

vector<POINT> lespoints; // variable globale

LRESULT CALLBACK MaMouseProc(int nCode, WPARAM wParam, LPARAM lParam)
{
    if(nCode < 0)
    {
        return CallNextHookEx(HookHandle, nCode, wParam, lParam);
    }
    else
    {
        if(wParam == WM_MOUSEMOVE)
        {
            MOUSEHOOKSTRUCT *mhs;
            mhs = reinterpret_cast<MOUSEHOOKSTRUCT*>(lParam);
            lespoints.push_back(mhs->pt);
            return true;
        }
        else
        {
            return CallNextHookEx(HookHandle, nCode, wParam, lParam);
        }
    }
}
```

Comme précédemment, nous n'intervenons que si nCode nous y autorise. wParam contient le type exact du message. Nous n'intervenons que lorsque ce type est WM_MOUSEMOVE. Les données relatives à l'événement sont rangées dans une structure MOUSEHOOKSTRUCT pointée par lParam. Le reste est simple. Voici maintenant le bout de code extrait de notre procédure de fenêtre pour

rejouer les événements à posteriori. Le principe est là encore tout simple, on constitue des événements à partir des données stockées dans un vector, puis on les injecte dans la file de l'application au moyen de l'API SendMessage :

```
case WM_TIMER:
    // Arrêter le timer
    KillTimer(hWnd, timer);
    // Enlever le hook
    unhooked = UnhookWindowsHookEx(HookHandle);
    if(!unhooked)
    {
        MessageBox(hWnd, L"Oups Unhook", L"Oups!", MB_OK);
        PostQuitMessage(0);
        break;
    }
    // et maintenant on rejoue les évènements WM_MOUSEMOVE
    for(vector<POINT>::iterator it = lespoints.begin();
        it!=lespoints.end(); ++it)
    {
        ScreenToClient(hWnd, &(*it));
        SendMessage(hWnd, WM_MOUSEMOVE, 0,
            MAKELPARAM((*it).x, (*it).y));
    }
    // Vider le vector pour une éventuelle
    // prochaine utilisation
    lespoints.clear();
    break;
```

4 INSTALLER UN HOOK GLOBAL

Si vous essayez notre exemple suivant, HookMouseInSystem, vous constaterez que celui-ci tue cette fois totalement la souris, pour toutes les applications. Ceci parce que nous installons un hook qui est global, et aussi parce que ce hook est du plus bas niveau possible. Ceci signifie qu'il intervient avant même que le message ne soit posté dans quelque file que ce soit. Un message supprimé par un tel hook n'atteint plus personne. Sur le principe le code de notre exemple ressemble beaucoup aux autres. Mais la difficulté réside ici dans l'écriture d'une DLL et du chargement de celle-ci par notre application principale. D'abord, il faut que l'application "voie" la librairie. Pour cela le plus simple est de configurer votre environnement de développement afin qu'il dépose les binaires de l'application et de la librairie dans un même répertoire. Ensuite il faut que la fonction du hook puisse être chargée dynamiquement. En voici le code :

```
extern "C" __declspec(dllexport) LRESULT CALLBACK MaLowLevel
MouseProc(int nCode, WPARAM wParam, LPARAM lParam)
{
    if(nCode < 0)
    {
        return CallNextHookEx(NULL, nCode, wParam, lParam);
    }
    else
    {
        return true;
    }
}
```

La déclaration extern "C" demande au compilateur C++ de ne pas décorer le nom de la fonction et __declspec(dllexport) lui demande de l'exporter. Et ... cela ne fonctionne pas! Sous Visual Studio 2008 du moins. Le compilateur s'obstine à décorer le nom de fonction comme on peut le voir sous Depends, un vieux outil Microsoft, hélas en voie de disparition [Fig.1]. Ceci sous prétexte, probablement, que notre hook obéit à la convention d'appel __stdcall [Fig.2]. Le plus simple pour remédier à ce problème est d'ajouter au projet un petit fichier texte, dit de définition, à l'extension .def, qui permet de définir des synonymes pour les noms de fonctions. Voici le contenu de notre fichier :

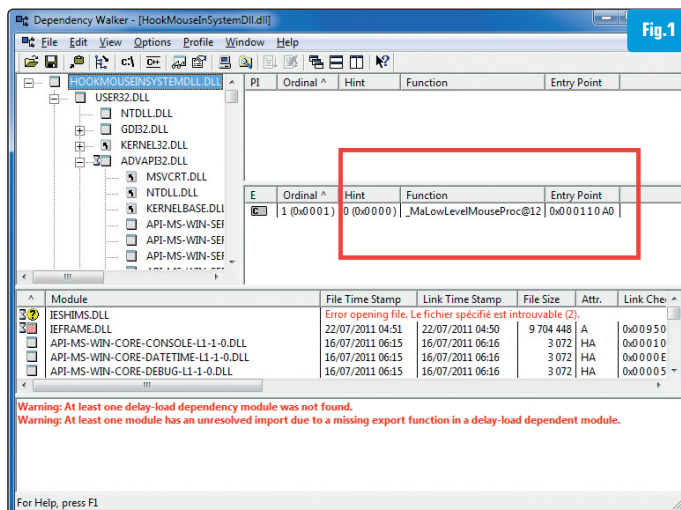
```
LIBRARY "HookMouseInSystemDll"

EXPORTS
MaLowLevelMouseProc = _MaLowLevelMouseProc@12
```

Un examen de nouveau code binaire produit par une recompilation montre que cette fois tout va bien se passer [Fig.3]. Voici maintenant, tiré de notre exemple, la partie du code qui charge la librairie et qui obtient l'adresse de notre procédure de hook. Le reste de l'exemple est identique aux précédents :

```
static HINSTANCE InstanceDll;
static HHOOK HookHandle;
FARPROC WINAPI HookAddr;

case ID_FILE_GO:
    // Installer le hook des évènements souris
    // D'abord charger la Dll contenant
    // la procédure de hook
    InstanceDll = LoadLibrary(L"hookmouseinsystemdll.dll");
    if(InstanceDll == NULL)
    {
        MessageBox(hWnd, L"Oups dll", L"Oups!", MB_OK);
        PostQuitMessage(0);
        break;
    }
    // Puis accéder à la procédure elle-même
    // et installer le hook
```



Oups, Visual C++ 2008 décore les noms de fonctions à l'insu de notre plein gré.

```
HookAddr = GetProcAddress(InstanceDll, "MaLowLevelMouseProc");
if(HookAddr == NULL)
{
    GetLastError();
    MessageBox(hWnd, L"Oups hook addr", L"Oups!", MB_OK);
    PostQuitMessage(0);
    break;
}
HookHandle = SetWindowsHookEx(WH_MOUSE_LL,
    reinterpret_cast<HOOKPROC>(HookAddr),
    InstanceDll, NULL);
```

5 REJOUER DANS UNE FENÊTRE, DES ÉVÉNEMENTS PRODUITS AILLEURS

Essayez notre dernier exemple, ReplayMouseInSystem. Cliquez sur le menu 'Go!' puis promenez votre souris par exemple un peu au dessus de la barre de titre de l'application. Ce faisant, vous ne générerez que des événements concernant le bureau ou, éventuellement d'autres applications, mais pas la nôtre. Pourtant, à l'expiration du timer, les événements sont (re)joués dans notre fenêtre. Le principe est tout simple, nous interceptons des événements avec un hook global, nous les enregistrons, puis nous les bricolons pour les rejouer localement. Il y a toutefois de petites difficultés de programmation, au niveau des pointeurs de fonction notamment. Voici le code de notre nouvelle dll :

```
#include <vector>

using namespace std;
vector<POINT> lespoints;

extern "C" __declspec(dllexport) LRESULT CALLBACK MaLowLevel
MouseProc(int nCode, WPARAM wParam, LPARAM lParam)
{
    if(nCode < 0)
    {
        return CallNextHookEx(NULL, nCode, wParam, lParam);
    }
    else
    {

```

```
if(wParam == WM_MOUSEMOVE)
{
    MOUSEHOOKSTRUCT *mhs;
    mhs = (MOUSEHOOKSTRUCT *)lParam;
    lespoints.push_back(mhs->pt);
    //return true;
    return CallNextHookEx(NULL, nCode, wParam, lParam);
}
else
{
    return CallNextHookEx(NULL, nCode, wParam, lParam);
}
}
```

```
extern "C" __declspec(dllexport) vector<POINT> GetHookedPoints()
{
    vector<POINT> temp;

    temp = lespoints;
    lespoints.clear();
    return temp;
}
```

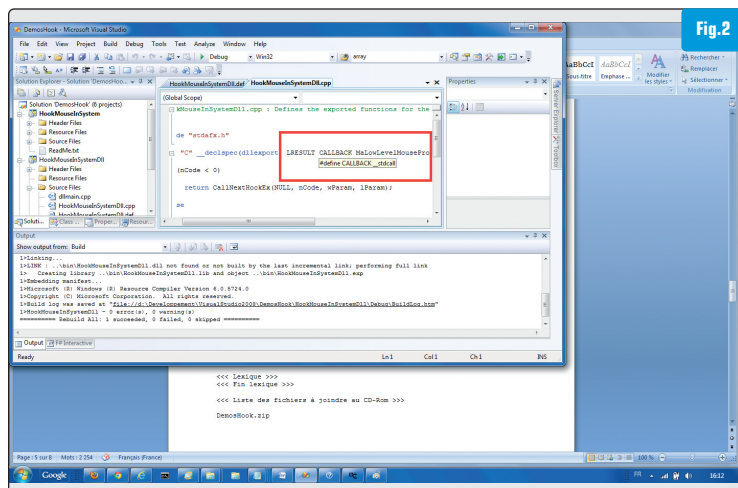
Nous notons l'apparition d'une nouvelle fonction, GetHookedPoints, dont le code permet de récupérer le vector contenant les événements enregistrés. La question est: comment invoquer une telle fonction quand la librairie est chargée dynamiquement ? La réponse est : avec un pointeur de fonction :-). Nous devons d'abord prototyper le pointeur de fonction ainsi :

```
typedef vector<POINT> (*MAFONCTION)();
```

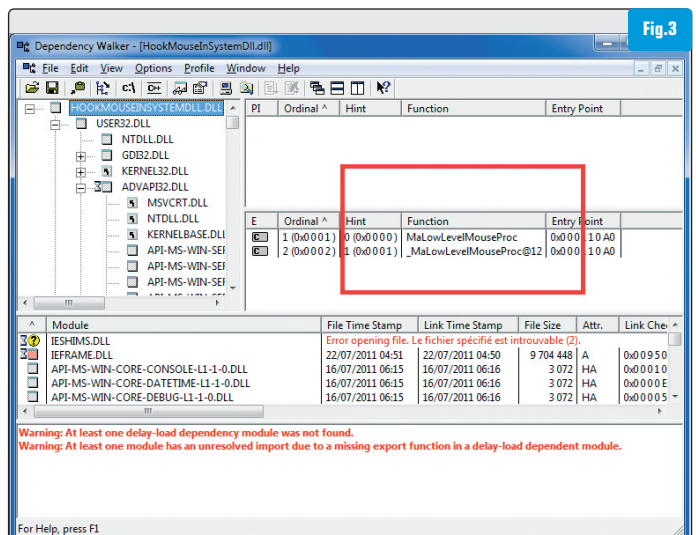
Le lecteur trouvera sur notre site le code qui charge la librairie, et qui obtient les adresses des fonctions depuis celle-ci.

Bien y remarquer static MAFONCTION MaFonction qui est affecté par l'adresse (i.e le pointeur) de la fonction GetHookedPoints. Ensuite à l'expiration du timer, nous invoquons tout simplement notre fonction avec MaFonction() pour récupérer les points enregistrés.

Frédéric Mazué - fmazue@programmez.com



La convention d'appel __stdcall des hook semble expliquer le comportement du compilateur.



Kinect SDK : éclate-toi !

C'est le 4 novembre 2010 que Microsoft a lancé pour sa Xbox 360 un nouveau périphérique de jeu. Avec le slogan « La manette c'est vous », ce périphérique se veut révolutionnaire. Derrière cette accroche se trouve un périphérique innovant permettant de jouer avec son corps, Kinect est née ! La promesse est grande. La petite caméra de Microsoft trouve un écho plus que favorable.

La vraie révolution ne viendra cependant pas de la Xbox 360 mais de la communauté qui se crée autour de Kinect. Il n'a pas fallu longtemps pour que certains s'approprient le périphérique en créant des frameworks pirates pour pouvoir utiliser Kinect avec un PC et, dans certains cas, avec d'autres objets.

Au lieu d'empêcher ce mouvement, Microsoft décide de supporter cette nouvelle communauté et nous gratifia le 16 mai 2011 d'un Kinect SDK pour que tout un chacun puisse développer ses applications. Le SDK permet de développer des applications mais pas de les vendre, dans ce cas précis, un SDK « commercial » arrivera courant 2012.

> Architecture Kinect

Avant toute chose il faut comprendre comment fonctionne Kinect. Ce périphérique est composé :

- D'un moteur qui fait bouger les caméras sur l'axe vertical, cela permet notamment à Kinect de pouvoir ajuster son champ de vision et se caler plus précisément sur les personnes à reconnaître.
- D'une Led qui peut prendre plusieurs couleurs (Rouge, Vert, Jaune). Elle se trouve devant et généralement indique l'état de Kinect.
- D'un « Microphone Array ». Concrètement, il s'agit de quatre micros qui permettent une détection du son dans l'espace, mais surtout de le capter en éliminant le bruit ambiant pour ainsi capter un flux audio propre et utilisable pour la détection de la voix par exemple.
- Une caméra VGA [Fig.1], qui permet de capturer une image vidéo standard en 640x480 ou 1280x1024.
- Un projecteur Infra-Rouge pour la détection des personnes. Ce projecteur affiche une grille infrarouge avec des points qui permettront ensuite d'être détectés par une caméra Infra Rouge et de déterminer la profondeur des objets.
- Une caméra Infra-Rouge [Fig.2], qui va capter la grille du projecteur et ensuite produire une image de profondeur. Comme la caméra utilise le projecteur, c'est une détection active, il faut entendre par là qu'il ne faut pas perturber le signal du projecteur (chaleur intense, autre capteur actif dans la pièce par exemple), ce qui aurait pour résultat d'induire des erreurs de détection.

Le capteur Kinect est donc bien outillé pour pouvoir capter nos mouvements, que l'on soit dans une pièce ou à l'extérieur. Il faut savoir que la détection d'une personne se situe entre une distance de 1,8



mètre à 3 mètres et que les personnes doivent avoir une taille supérieure à 1 mètre. En ce qui concerne l'architecture [Fig.3] applicative de Kinect, le Kinect SDK (développé par Microsoft Research) permet de créer des applications en managé et C++ (x86 et x64) avec Visual Studio 2010. Ces applications ne peuvent fonctionner que sous Windows 7. Il est à noter que Kinect n'embarque pas directement les algorithmes de reconnaissance, ce travail est fait par le PC (ou la Xbox), Kinect ne fait que relayer les informations vers le SDK.

> Interactions

Maintenant que l'on sait ce que Kinect a dans le ventre, nous allons nous intéresser à ce que le capteur peut nous fournir en termes d'information et ce qu'il peut faire pour nous. Nous allons distinguer deux grands types d'interactions, la voix et la reconnaissance des personnes.

Les mouvements

C'est sans doute la possibilité la plus utilisée et la plus intéressante de Kinect. Nous avons vu précédemment qu'il y a deux flux retournés par Kinect, une image vidéo VGA et une image de profondeur. A partir de ces deux flux, Kinect est capable de nous donner deux informations supplémentaires :

- La détection de la présence de quatre personnes et nous fournir leurs ombres (nous y reviendrons plus tard)

- Sur deux de ces personnes, Kinect peut suivre leur squelette [Fig.4]. Kinect va pour cela observer 20 points, ou jointures, afin d'informer des mouvements des utilisateurs.

Le son

Kinect est capable de faire de la reconnaissance vocale mais pas nativement. En fait, il fournit un flux audio primitif qu'il est possible de faire interpréter par des frameworks comme Microsoft Speech par exemple. L'avantage de la prise de son Kinect, est qu'il lui est possible de supprimer le bruit ambiant pour fournir un flux plus propre et plus en adéquation pour de la détection ou du traitement de signal, mais aussi de détecter d'où vient le son (positionnement dans l'espace).

Il est donc possible de piloter son interface avec la voix, ce qui permet, notamment dans le jeu vidéo, de pouvoir donner des ordres à des personnages non joueurs comme dans Mass Effect 3, ou accélérer, ralentir et mettre en pause le jeu comme dans Dance Central 2.

De manière générale, il est toujours préférable de dire un mot clé avant la commande vocale (comme le mot « xbox ») afin que des détections intempestives ne soient pas trouvées suite à une discussion normale.

Usages

Bien que Kinect ait été conçu et vendu pour le jeu, de nombreux usages que Microsoft n'avait pas envisagés apparaissent. On peut classer ces nouveaux usages en deux catégories. Des usages ludiques où l'on va utiliser Kinect pour des campagnes de marketing, des événements autour d'une marque par exemple. Des usages plus sérieux avec une utilisation de Kinect pour aider les personnes à mobilité réduite en entreprise, à la fois pour les employés mais aussi pour les clients.

> Utilisation du SDK

On utilise le SDK Kinect avec Visual Studio 2010, peu importe la version. Je vais aborder ici comment utiliser le SDK en mode managé (C#) mais il est également possible de l'utiliser en C++. Tout d'abord voyons quelques bases sur le SDK.

Le Runtime

Lorsque l'on veut créer un projet Kinect, on doit ajouter l'assembly « Microsoft.Research.Kinect ». Ensuite on accède aux API par deux namespaces : « Microsoft.Research.Kinect.Nui » et « Microsoft.Research.Kinect.Audio » qui permettent respectivement de gérer la détection vidéo et la gestion de la prise de son.

La première classe intéressante est statique et se nomme « Runtime », elle donne accès au nombre de Kinect actuellement connectés

via la propriété « Kinects » et permet aussi de se brancher sur un événement nous indiquant quand on connecte / déconnecte un Kinect, où lorsque l'un d'eux tombe en erreur. C'est très pratique pour avoir un état des lieux très précis des périphériques connectés.

```
public MaFenetre()
{
    // Récupère le nombre de Kinect actuellement connectés
    this._kinectCount = Runtime.Kinects.Count;

    // On se branche sur le scrutter d'état des Kinects
    Runtime.Kinects.StatusChanged += new System.EventHandler<Status
    ChangedEventArgs>(this.KinectsStatusChanged);
}

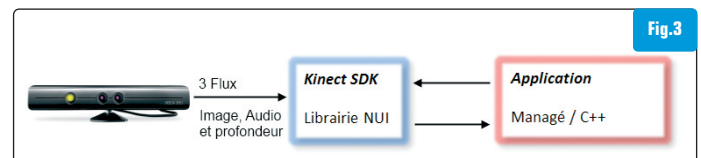
private void KinectsStatusChanged(object sender, StatusChanged
    EventArgs e)
{
    // Ici on regarde le statut, si le Kinect est connecté ou
    // déconnecté
    if (e.Status == KinectStatus.Connected)
        this._kinectCount ++;
    else if (e.Status == KinectStatus.Disconnected)
        this._kinectCount --;
}
```

Le plus intéressant, c'est que sur l'événement on récupère (dans le cas d'une connexion), l'instance du « Runtime » Kinect correspondant au périphérique connecté (par la propriété « KinectRuntime »), il n'y a plus qu'à l'utiliser tel quel pour interagir avec celui-ci.

Si l'on veut récupérer l'instance d'un « Runtime » déjà existant, il faut aller piocher dans la collection « Kinects » de l'objet statique « Runtime » (Microsoft.Research.Kinect.Nui).

Vidéo et détection des mouvements

Maintenant que nous avons l'instance « Runtime » qui représente notre Kinect, si nous souhaitons utiliser la reconnaissance de mouvement, il faut démarrer cette reconnaissance et se brancher sur



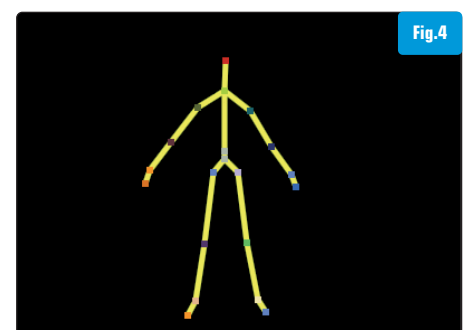
Architecture



Image VGA



Image de profondeur



Squelette capturé

ses événements. Mais avant toute chose il faut initialiser le « Runtime » pour lui indiquer ce qu'il doit traiter.

```
this._runtime.Initialize(RuntimeOptions.UseColor | RuntimeOptions.
UseDepthAndPlayerIndex | RuntimeOptions.UseSkeletalTracking);
```

En fonction des paramètres passés dans la fonction « Initialize », Kinect va envoyer les informations demandées. Le « UseColor » pour la caméra vidéo, le « UseDepth » et « UseDepthAndPlayerIndex » pour la caméra de profondeur et enfin « UseSkeletalTracking » si vous voulez récupérer les informations de squelette.

Pour démarrer un flux vidéo il faut passer par la méthode « Open » de « VideoStream », pour démarrer la caméra de profondeur il faut passer par la méthode « Open » de « DepthStream ».

Une fois les flux démarrés (il n'est pas obligatoire de faire les « Open » pour récupérer les informations de trace liées au squelette), il suffit de s'abonner aux événements « VideoFrameReady » (VideoStream.Open nécessaire), « DepthFrameReady » (DepthStream.Open nécessaire) et « SkeletonFrameReady » (« Runtime.SkeletonEngine.IsEnabled » doit être activé) pour récupérer les informations que nous renvoie Kinect.

```
this._runtime.Initialize(RuntimeOptions.UseColor | RuntimeOptions.
UseDepthAndPlayerIndex | RuntimeOptions.UseSkeletalTracking);
```

```
this._runtime.VideoStream.Open(ImageStreamType.Video, 2, Image
Resolution.Resolution640x480, ImageType.Color);
this._runtime.DepthStream.Open(ImageStreamType.Depth, 2, Image
Resolution.Resolution320x240, ImageType.DepthAndPlayerIndex);
```

```
this._runtime.DepthFrameReady += new EventHandler<ImageFrameReady
EventArgs>(this.DepthFrameReady);
this._runtime.SkeletonFrameReady += new EventHandler<SkeletonFrame
ReadyEventArgs>(this.SkeletonFrameReady);
this._runtime.VideoFrameReady += new EventHandler<ImageFrameReady
EventArgs>(this.VideoFrameReady);
```

Il est à noter que pour certains flux les résolutions sont imposées, pour le flux vidéo c'est obligatoirement 1280x1024 et 640x480, pour le flux de profondeur c'est 320x200 et 80x60. Comme l'énumération est identique, attention à ne pas se tromper. Il en va de même pour les « ImageType », certaines valeurs de l'énumération sont réservées pour la vidéo (Color, ColorYuv, ColorYuvRaw) et d'autres pour la profondeur (Depth, DepthAndPlayerIndex).

Sur le callback des événements, les informations Vidéo et de profondeur seront retournées sous forme de tableau d'octet (PlanarImage). Il faudra donc les convertir pour les afficher. Ci-dessous les méthodes pour l'affichage du flux vidéo dans une fenêtre WPF.

```
private void VideoFrameReady(object sender, ImageFrameReady
EventArgs e)
{
    PlanarImage image = e.ImageFrame.Image;

    BitmapSource source = BitmapSource.Create(image.Width, image.
Height, 96, 96, PixelFormats.Bgr32, null, image.Bits, image.
```

```
Width * image.BytesPerPixel);
    this.VideoBrush = new ImageBrush(source);
}
```

Pour l'image de profondeur, c'est un peu plus compliqué. Il faut tout d'abord savoir que l'image de profondeur nous renvoie pour chaque pixel 2 octets, soit 16 bits d'information. Dans ces 16 bits, seulement 13 sont utilisés, 3 bits pour renvoyer l'index du joueur détecté (jusqu'à 4, 1 étant le plus près, 4 le plus loin et 0 si pas de joueur détecté) et enfin 10 bits pour la distance en millimètre qui sépare le premier obstacle rencontré par la caméra infrarouge sur ce pixel par rapport à Kinect (qui est le point 0). Pour récupérer ces informations il faut donc faire quelques manipulations :

```
/// <summary>
/// Récupère la distance renvoyée par Kinect pour un pixel
/// </summary>
/// <param name="byte1">Premier octet (distance + joueur)</param>
/// <param name="byte2">Deuxième octet (distance + 3 bit mort)</param>
/// <returns>Distance calculée</returns>
private int GetDistance(byte byte1, byte byte2)
{
    return ((int)(byte1 >> 3 | byte2 << 5));
}

/// <summary>
/// Récupère l'index du joueur
/// </summary>
/// <param name="byte1">Octet a tester (distance + joueur)</param>
/// <returns>Index du joueur, 0 si pas de joueur</returns>
private int GetPlayerIndex(byte byte1)
{
    // En binaire 7 s'écrit : 00000111. Cela nous permet de récupérer les
    // 3 derniers bits de notre octet et donc récupérer l'index du joueur
    return ((int)( byte1 & 7));
}
```

Pour afficher l'image de profondeur il faut donc se créer un buffer et traiter un à un les 2 octets renvoyés, il vous appartient ensuite d'interpréter la distance et l'index du joueur pour générer une image en niveau de gris, voir coloriser les joueurs.

Maintenant voyons comment gérer la détection du squelette. Ce qui est facile avec le SDK Kinect c'est que toutes les informations liées à la détection du squelette sont envoyées dans le callback de l'événement « SkeletonFrameReady ». Dans les arguments de l'événement il est possible de récupérer un objet « SkeletonFrame » qui contient toutes les informations du squelette à un instant du callback. C'est avec cet objet qu'on peut traiter le mouvement, par exemple afficher le squelette en mode fil de fer ou essayer de détecter des gestuelles en regardant les mouvements sur un laps de temps.

Il y a plusieurs informations intéressantes à récupérer de ce « SkeletonFrame », tout d'abord la liste des squelettes, car je vous rappelle que Kinect peut tracer six personnes. Via la propriété « SkeletonFrame.Skeletons » on obtient la liste des squelettes sous la forme de « SkeletonData ».

Dans le « SkeletonData » il est possible d'accéder à l'index du joueur (UserIndex), mais pour savoir si ce squelette est tracé avec les 20 points ou juste avec son centre de gravité, il faut interroger la propriété « TrackingState » qui vous dit si oui ou non ces informations sont disponibles. Ce « TrackingState » vous indique que sur les six personnes maximum que peut gérer Kinect, quatre seront en statut « NotTracked », c'est-à-dire que Kinect ne suit pas réellement ces personnes. Deux seront en statut « PositionOnly », c'est-à-dire que Kinect ne suit que leur centre de gravité. Enfin, les deux dernières personnes seront en statut « Tracked », c'est-à-dire que des informations précises sur les 20 articulations suivies sont disponibles. Cela se produira dans le cas où vous avez six personnes devant Kinect, si seulement trois personnes sont devant la caméra vous avez deux personnes en « Tracked » et une personne en « PositionOnly », mais toujours six « SkeletonData » dans la collection « SkeletonFrame.Skeletons ».

Ce qui est intéressant avec le SDK, c'est qu'il vous renvoie une valeur de fiabilité sur sa détection. Ainsi, sur le « SkeletonData » vous avez une propriété « Quality » (entre 0 et 1, 1 étant parfait) qui vous donne l'information sous forme de pourcentage de la probabilité de la détection, on peut estimer qu'au-delà de 80% (0.8), la détection du squelette est exploitable et de bonne qualité.

Si l'on veut récupérer le centre de gravité du squelette, c'est la propriété « Position » qu'il faut récupérer. Vous pouvez aussi récupérer le « UserIndex » qui est au final une information peu utile, elle vaut toujours 254 si on est en mode squelette. Enfin le « TrackingID » qui, lui, donne plutôt un index persistant entre chaque « Frame » pour chaque personne en cours de détection, tant que la personne reste devant Kinect le « TrackingID » ne change pas, par contre si la personne revient elle possèdera un nouveau « TrackingID ».

Un autre aspect intéressant lorsque l'on fait du suivi de squelette réside dans le fait que Kinect peut appliquer des algorithmes pour affiner la détection et réduire le bruit sur cette dernière. Kinect peut appliquer deux stratégies (Correction et/ou lissage) sur les données qu'il capture. Pour activer cette correction, il faut activer sur le « Runtime » la propriété « SkeletonEngine.TransformSmooth » à « True » et ensuite régler les paramètres de filtrage par la propriété « SkeletonEngine.SmoothParameters ».

```
TransformSmoothParameters parameters = new TransformSmoothParameters();
parameters.Correction = 0.5f;
parameters.Prediction = 0.5f;
parameters.Smoothing = 0.05f;
parameters.JitterRadius = 0.05f;
parameters.MaxDeviationRadius = 0.04f;
```

```
this._runtime.SkeletonEngine.SmoothParameters = parameters;
this._runtime.SkeletonEngine.TransformSmooth = true;
```

Maintenant que l'on sait activer la détection du squelette, que l'on sait comment éliminer le bruit, il est temps de passer à l'interprétation des données que nous renvoie Kinect sur ce suivi.

Tout se passe avec la propriété « Joints » du « SkeletonData », cette collection renvoie les 20 points de détection avec leurs coordonnées et leur état. En effet, l'état de détection d'un « Joint » est très important. Prenons la main droite, si on la met face à Kinect, l'état (TrackingState) du « Joint » est à « Tracked », cela veut dire qu'il n'y a pas de problème, Kinect le gère sans souci. Par contre, si on met la main derrière son dos, impossible pour Kinect de savoir où elle est, alors le « TrackingState » passe à « NotTracked », ce qui permet de gérer très précisément les 20 points de suivi.

Il existe un autre état qui est « Inferred », cela veut dire que Kinect en a déduit sa position mais ce n'est pas le résultat d'une détection directe. Cela peut arriver quand vous collez vos deux mains, Kinect sait que les mains sont là (déduit des positions précédentes) mais n'a pas réussi à les détecter directement avec les senseurs. A ce moment-là, il vous indique que c'est une déduction.

Pour récupérer un « Joint » précis, vous avez à votre disposition une énumération nommée « JointID » qui donne la liste de

tous les « Joints » avec des mots en anglais. C'est plus simple que de se souvenir des codes numériques. La collection « Joints » peut bien sûr être indexée par cette énumération.

Sur les « Joints », vous avez accès à la propriété « Position » qui vous donne comme son nom l'indique, sa position dans l'espace. C'est une structure Vecteur qui donne les informations X, Y, Z et W. Oui, il y a bien une quatrième information, car dans l'espace seul X, Y et Z suffisent pour se positionner. W représente, à l'instar de la « Quality », la qualité de détection de la position, cela s'exprime entre 0 et 1, 1 étant une détection parfaite. Cela est très utile si on ne veut garder que des « Joints » supérieurs à 80% de détection et ainsi éliminer les détections parasites. Vous pouvez maintenant, afficher le squelette dans un Canvas WPF par exemple, en créant des lignes et des points pour faire rejoindre les informations des « Joints ».

En conclusion, ceci est une première approche des possibilités incontournables du SDK Kinect, à vous maintenant de laisser libre cours à votre imagination.



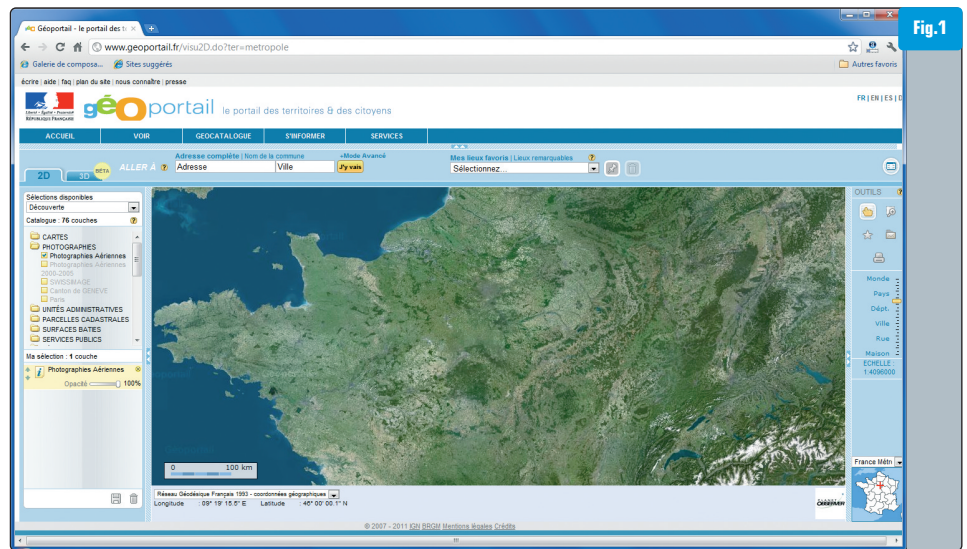
Nicolas Calvi
MVP Surface
Consultant / Formateur - Société Winwise

OFFRE SPÉCIALE

ABONNEMENT : UN LIVRE NUMÉRIQUE ENI OFFERT (voir page 34)

Mettre en œuvre l'API Géoportail

Le Géoportail est un service qui permet de visualiser les données géographiques et cartographiques françaises et mondiales via son site officiel. Mais il met aussi une API à disposition, pour vous permettre d'intégrer ces données dans votre site. Découverte.



Les services du Géoportail, tels que présentés par le site officiel.

Au départ, le Géoportail est un portail Web français [Fig.1] qui a été lancé en 2006, et dont le but est de permettre à chacun d'accéder gratuitement aux données géographiques de tout le territoire français, à www.geoportail.fr. Ces données étant à la base constituées de photographies aériennes et de données cartographiques. Depuis cette date, c'est un outil en constante évolution. Son champ d'application s'est étendu au monde entier, même si le territoire français, fort logiquement, reste son terrain de prédilection. Au fil du temps le Géoportail a intégré d'autres données en provenance de partenaires, comme des images satellite ou autres. Il propose maintenant une API qui vous permettra d'afficher ces données dans votre site Web, ceci gratuitement si votre site est sans but lucratif. Il vous permettra aussi de superposer des couches de données de provenances tierces.

1 GÉOPORTAIL VS GOOGLE MAPS

Relativement peu connu, Géoportail se présente comme un concurrent de Google Maps, qui est, lui, très connu. Tous les deux ont des buts semblables : montrer des cartes dans un navigateur. Mais ceci avec des philosophies différentes qui leur confèrent leurs spécificités. D'entrée de jeu, Google Maps couvrait le monde entier, mais de manière très inégale. Les lieux les plus "intéressants" étant couverts avec une meilleure résolution que les lieux "quelconques". Une situation qui évolue petit à petit, car Google ne cesse d'intégrer de nouvelles données à son service. Des images satellites bien sûr, mais aussi des données de services tiers comme l'IGN (Institut Géographique National) et donc finalement du Géoportail :) Celui-ci, de son côté, a d'abord exposé les données du territoire français dans son entier, avec une résolution identique en tout lieu, selon un principe "d'égalité et de satisfaction de l'intérêt général". Ces données sont principalement constituées des cartes au 1 / 25 000, anciennement dites cartes d'états major de l'IGN, et de photographies aériennes, toujours en provenance de l'IGN, qui sont d'une résolution de 50cm ou mieux. Petit à petit, le Géoportail intègre lui aussi des données tierces, comme des photographies satellites, et il couvre maintenant

le monde entier. Les deux services tendent donc de plus en plus à se ressembler et on peut se demander lequel choisir ? Google Maps présente un côté plus grand public et il ne connaît qu'un seul système de coordonnées: les latitudes et longitudes. Géoportail, quant à lui, présente un aspect plus professionnel, plus géographe. En ce qui concerne les coordonnées, outre les classiques latitudes et longitudes, le Géoportail en connaît de nombreux autres, RFG93, Lambert, etc. Tous les systèmes de coordonnées du standard OGC sont connus par le Géoportail. En outre celui-ci se différencie nettement avec des données uniques comme les couches cadastrales, ou les bâtiments en 3D. Mais sa prise en main est un petit peu plus ardue que celle de Google Maps, et sa documentation n'est pas toujours limpide.

2 PRÉREQUIS, OUTILS ET CONTRATS

Géoportail propose deux jeux d'API: en Flash et en Javascript. Le premier propose un viewer [Fig.2] beaucoup plus fluide que le second, mais qui nécessitera peut être pour vous d'apprendre le standard KML pour passer des données au lecteur Flash. L'API Javascript présente un viewer moins fluide, mais elle est plus évolutive. Basée sur la librairie open source OpenLayers, elle évolue au rythme de cette dernière, ainsi que grâce aux travaux de l'IGN. Le code source de cette API est sous licence BSD. Nous nous intéresserons essentiellement à l'API Javascript dans cet article. Quelle que soit l'API que vous choisirez, vous avez besoin d'un serveur HTTP pour essayer votre code. Pour le développement en local sous Windows, le serveur Apache d'EasyPHP conviendra par exemple très bien. Vous devez ensuite souscrire un contrat auprès du Géoportail. L'opération consiste essentiellement en la création d'un compte à <http://api.ign.fr/geoportail>. Ensuite vous devez sélectionner un contrat selon l'usage, commercial ou non, que vous souhaitez faire de l'API. Pour le développement en local (avec le serveur http qui donc tourne sur votre machine de développement) et les essais, Choisissez un contrat de type "L'API Géoportail dans un site web non commercial", puis dans le formulaire qui vous sera présenté, spécifiez <http://localhost> comme URL de votre site. Cochez également toutes les couches de

données que vous souhaitez accessibles pour ce contrat. N'hésitez pas: pour vous familiariser avec l'API, cochez tout :) A l'issue de l'opération, il vous sera attribué une clé de contrat que vous devrez systématiquement passer à l'API à chaque chargement d'une page Web.

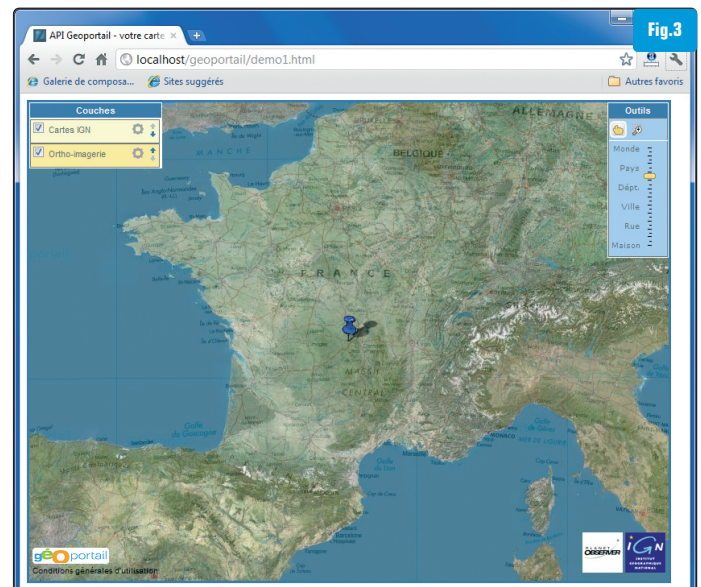
3 HELLO WORLD !

Le site dédié aux API Géoportail donne un exemple de type Hello World pour travailler avec le lecteur Flash. Vous trouverez ce code dans `demoflash.html` sur notre site. Nous ne reviendrons pas sur ce code, mais signalons au passage une erreur dans la documentation. Celle-ci indique de télécharger un fichier `.swf`, puis indique que ce fichier doit être placé dans un répertoire dédié du nom de `APIFlash`. Cette indication est erronée. Il faut et il suffit de placer ce fameux fichier conjointement à vos fichiers HTML. Venons en maintenant à la programmation Javascript. Le lecteur trouvera sur notre site `demo1.html` qui est une page HTML de base proposée, par Géoportail, et qui affiche la carte de France [Fig.3] dans le viewer.

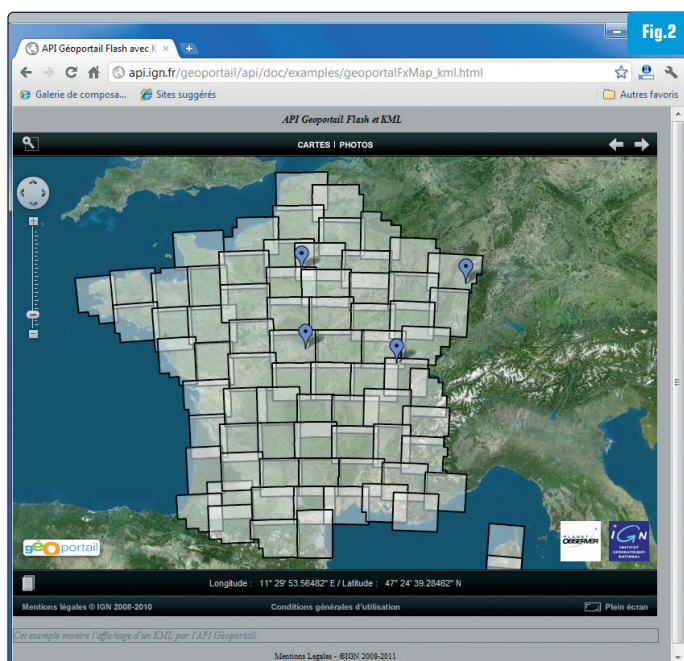
Ce code affiche le viewer dans une DIV HTML. Nous voyons que nous passons notre clé de contrat à l'API. Nous voyons aussi que nous passons la latitude et la longitude du point sur lequel la carte sera centrée. Mais ce que nous ne voyons pas, c'est pourquoi notre affichage présente deux couches (carte IGN et photographies aériennes), ni avec quelle partie de l'API nous travaillons exactement. C'est le moment d'aller faire un petit tour dans la documentation, à <http://api.ign.fr/geoportail/api/doc/fr/index.html>. Une fois là, cliquez sur le lien `Geoportal JS 1.2` ce qui vous amène à la documentation de la librairie Javascript. N'oubliez pas que cette librairie est construite sur `OpenLayers`, ainsi si le fonctionnement de certaines classes vous paraît peu clair, vérifiez bien que la classe en question ne dérive pas d'une classe `OpenLayers`, et si oui, rendez-vous à sa documentation qui est séparée. Dans la présente documentation, cliquez sur 'Viewer' qui est le sujet qui nous intéresse', puis à nouveau dans le sous-lien 'Viewer'. Tout à fait en bas de la page se cache la fonction utilitaire `Geoportal.load` utilisée par notre exemple. La documentation de cette

fonction nous explique comment il est possible de personnaliser le viewer, dans une certaine mesure. Ainsi dans l'exemple suivant (`helloworld.html`) nous réécrivons notre code pour avoir un viewer minimaliste qui n'affiche que la couche des photographies aériennes, et qui centre la carte sur Paris, [Fig.4] ce qui est aussi le comportement par défaut. Nous donnons enfin l'altitude de l'observation en spécifiant une valeur de `zoom`.

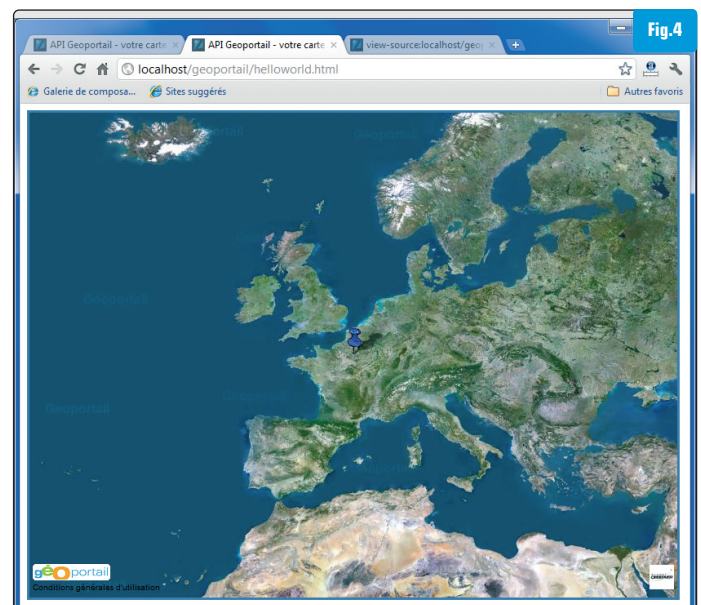
```
<script type="text/javascript"><!--//--><![CDATA[//><!--
window.onload= function() {
    Geoportal.load(
        // ID de la DIV qui va intégrer le viewer:
        'viewerDiv',
        // Clé de contrat:
        '6xxxxxxxxx4',
        // objet 'pos'.
        { // On se positionne sur Paris :
```



Le viewer, tel que chargé par le code de démonstration proposé par le Géoportail.



Le viewer Flash de l'API Géoportail.



Nous avons personnalisé le viewer pour lui donner un comportement minimaliste.


```
// ce qui est aussi la position par défaut
// longitude:
lon:2.37,
// latitude:
lat:48.84
},

// Valeur de zoom
2,
// objet options
{
  layers: ['ORTHOIMAGERY.ORTHOPHOTOS'],
  mode: 'mini',
  viewerClass: Geoportal.Viewer.Default,
});
};
//--><[!]]>
</script>
```

Mais nous avons aussi la désagréable surprise de voir s'afficher une punaise, qui apparaîtrait même si nous ne donnons pas le centrage de la carte. Impossible de s'en débarrasser. Elle est dessinée dans un layer ou couche spéciale, ce que nous verrons plus loin. En théorie il est possible de retirer un layer de l'affichage, si le code accède à l'objet correspondant. Mais votre serveur, lors de ses essais n'a pas trouvé comment faire. De toute façon, pour profiter pleinement de l'API, il est préférable de construire l'affichage pas à pas, ce que nous faisons maintenant.

4 CONSTRUIRE UN AFFICHAGE PAS À PAS

Le code ci-dessous (basic-0.html) montre l'autre approche pour instancier le viewer dans une page Web. Cette approche permet de travailler étape par étape.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<script type="text/javascript"
src="http://api.ign.fr/geoportal/api?v=1.2&key=6xxxxxxx
xxx4&instance=mymap&";>
</script>
<script type="text/javascript">
<!--
function initGeoportalMap() {
// Cette fonction sera automatiquement
// exécutée au chargement de la page HTML
// C'est ici que la carte doit être personnalisée
geoportalLoadmymap('mymapdiv', 'normal');
//mymap.addGeoportalLayers();
}
-->
</script>
</head>
<body>
```

```
<div id="mymapdiv" style="width:800px;height:600px;"></div>
</body>
</html>
```

Bien remarquer la nouvelle balise script. Elle demande le chargement de l'API en passant la clé de contrat au serveur du Géoportail, mais surtout elle passe le nom d'instance du layer qui sera ensuite utilisé dans tout le code. De quelle instance d'objet s'agit-il ? De `geoportal.viewer`, que nous avons rencontré plus haut. La deuxième balise script contient le code d'initialisation. Ce code doit par convention comporter une fonction de nom `initGeoportalMap`, qui sera appelée automatiquement. Dans ce code, la première chose à faire est d'appeler une fonction que l'API nous aura bricolée à partir du nom d'instance du viewer. Comme nous avons précédemment donné `mymap` comme nom de viewer, la fonction que nous devons appeler est :

`geoportalLoadmymap`

Si vous retirez les commentaires de la ligne suivante vous obtenez le viewer par défaut, avec les deux couches que nous avons rencontrées précédemment. Sans cela notre affichage est vide. Pour ajouter des couches à volonté, il suffit maintenant de connaître leur nom que l'on trouvera dans la documentation comme illustré [Fig.5]. Ainsi pour retrouver nos deux couches par défaut, le code est (basic-1.html) :

```
mymap.addGeoportalLayer('GEOGRAPHICALGRIDSYSTEMS.MAPS');
mymap.addGeoportalLayer('ORTHOIMAGERY.ORTHOPHOTOS');
```

5 COMPRENDRE LE MÉCANISME D'OPACITÉ DES COUCHES

Les couches se superposent et peuvent être toutes visibles en même temps par transparence. Pour arriver à cela, le principe, non expliqué, est simple. Les couches apparaissent dans le viewer dans l'ordre où le code les déclare. Et plus la valeur d'opacité est faible, plus la couche est transparente. Une couche peut être visible ou non, lors de l'ajout par le code :

```
mymap.addGeoportalLayer('GEOGRAPHICALGRIDSYSTEMS.MAPS',
{opacity: 0.5, visibility: false});
```

Fig.5

Identification des couches

Nom harmonisé des couches

Les couches principales sont identifiées par les codes suivants :

Code	Nom
ADDRESSES.CROSSINGS	Adresses début-fin sur les tronçons
ADMINISTRATIVEUNITS.BOUNDARIES	Limites administratives
BUILDINGS.BUILDINGS	Constructions
CADASTRALPARCELS.PARCELS	Parcelles cadastrales
ELEVATION.SLOPES	NT sous forme d'une image en te
GEOGRAPHICALGRIDSYSTEMS.MAPS	Cartes scannées, quelque soit leur INSPIRE
GEOGRAPHICALNAMES.NAMES	Noms de lieux
HYDROGRAPHY.HYDROGRAPHY	Réseaux hydrographiques
ORTHOIMAGERY.ORTHOPHOTOS	Ortho-photographies, quelque soit
ELEVATION.LEVELO	Traits de côte
TOPONYMS.ALL	Toponymes
TRANSPORTNETWORKS.RAILWAYS	Réseaux ferroviaires
TRANSPORTNETWORKS.ROADS	Réseaux routiers
TRANSPORTNETWORKS.RUNWAYS	Listes d'aéroports, d'aérodromes
UTILITYANDGOVERNMENTALSERVICES.ALL	Ouvrages et constructions gouvern

Ces codes sont basés sur la directive Européenne INSPIRE, annexes I à III à partir des noms cadre de l'application de la directive, les services fourniront les noms harmonisés prévus par d'avoir des noms homogènes dans toute l'Europe pour faciliter la mise en place des services.

Le nom des couches de données, dans la documentation.


```
mymap.addGeoportalLayer('ORTHOIMAGERY.ORTHOPHOTOS',
    {opacity: 0.5});
```

Ce code (basic-2.html) superpose cartes et photographies aériennes. Au départ seules ces dernières sont affichées, tant que l'utilisateur ne clique pas sur la case à cocher. Lorsqu'il n'y a qu'une couche affichée, le résultat est pâlichon et peu satisfaisant en raison de la faible opacité. Le résultat n'est guère mieux avec les deux couches simultanément. Il me semble plus intéressant de jouer avec des contrastes en donnant des opacités différentes, par exemple comme ceci (basic-3.html)

```
mymap.addGeoportalLayer('ORTHOIMAGERY.ORTHOPHOTOS',
    {opacity: 0.8});
mymap.addGeoportalLayer('GEOGRAPHICALGRIDSYSTEMS.MAPS',
    {opacity: 0.3});
```

6 SAVOIR TRAVAILLER AVEC LES COORDONNÉES

Nous pouvons maintenant centrer notre carte, par exemple sur Paris (basic-4.html)

```
themap = mymap.getMap();
themap.setCenterAtLonLat(2.37, 48.84, 12, false, false);
```

Nous voyons que le centrage doit être fait sur la carte détenue par le viewer. Les coordonnées sont passées en degrés décimaux. Il est aussi possible de passer les coordonnées en degrés, minutes, secondes, via une chaîne de caractères, en lieu et place des valeurs numériques. Le lecteur consultera avec profit la documentation des fonctions de conversion `degToDMS` et `dmsToDeg`, sous `Geoportal.Util`. Le point ardu est maintenant que le Géoportail utilise un très grand nombre de systèmes de coordonnées (plus de 200!), qui ne changent pas seulement en fonction du choix de l'utilisateur, mais aussi du lieu sur lequel on pointe. Pire, le viewer utilise toujours simultanément 2 systèmes de coordonnées qui ne sont jamais les latitudes et longitudes 'simples' :-). Les exemples `basic-5.html` et `basic-5.bis.html`, dont le code est identique, mais qui pointent sur Paris et la Guadeloupe respectivement, illustrent ceci. Supposons que ces exemples soient la base d'un outil qui permette de récupérer les coordonnées géographiques d'un lieu. Voici ce que cela donne:

```
function cliclistener(e) {
    themap = mymap.getMap();
    pos = themap.getLonLatFromViewPortPx(e.xy);
    alert(pos);
    pos.transform(themap.getProjection(), themap.displayProjection);
    alert(pos);
    alert(themap.getProjection());
    alert(themap.displayProjection);
}

themap = mymap.getMap();
themap.events.register("click", themap, cliclistener);
```

Ce code commence par récupérer les coordonnées dans l'affichage graphique du lieu, puis les affiche dans une boîte de dialogue. Ces coordonnées sont, disons, "pour la cuisine interne" du Géoportail. Le nom du système utilisé s'obtient par la fonction `getProjection` (Ici tout

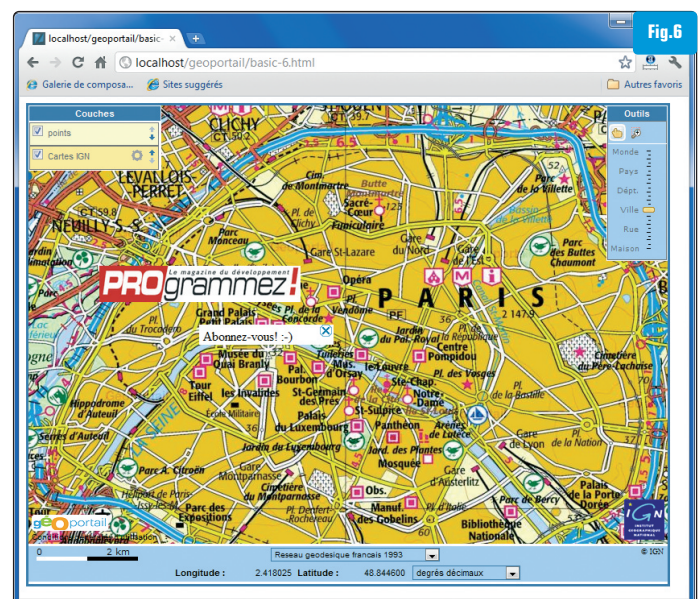
est documenté dans la documentation d'OpenLayers, et plus dans celle de Géoportail...) Nous les transformons ensuite en coordonnées d'affichage... C'est ici que la terminologie est totalement contre-intuitive pour un programmeur. Il s'agit de l'affichage des coordonnées à destination de l'utilisateur, dans la barre sous le graphique, et non l'affichage dans la carte. Nous affichons enfin la valeur des ces coordonnées, puis le nom des systèmes utilisés. Essayez nos exemples et appréciez comment les noms des systèmes varient selon le lieu :-). Nous pourrions aussi songer à un outil de transformations de coordonnées. Ainsi un utilisateur qui connaîtrait la latitude et la longitude d'un lieu pourrait se positionner dans le viewer et cliquer sur le lieu pour obtenir ses coordonnées Lambert. La base de code (basic-7.html) est la suivante :

```
function cliclistener(e) {
    themap = mymap.getMap();
    pos = themap.getLonLatFromViewPortPx(e.xy);
    pos.transform(themap.getProjection(),
        new OpenLayers.Projection('IGNF:LAMB93'));
    alert(pos);
}
```

7 AJOUTER SES PROPRES COUCHES

Il est possible de créer ses propres couches et de les ajouter dans le viewer. Il est encore possible d'ajouter des couches en provenance d'un serveur tiers. Dans tous les cas, le secret de l'opération réside dans un système de coordonnées identique pour toutes les couches :-). Basic-6.htm, que vous trouverez sur notre site ajoute le logo de **Programmez!** sur l'Arc de Triomphe (Ne nous refusons rien :-). sur la carte de Paris, ainsi qu'une bulle, dans une couche vectorielle. [Fig.6] Le point important de ce dernier exemple est que nous créons un layer dans le système de coordonnées par défaut sur Paris (pas de pixels ici :-). Quand le layer est ajouté dans le viewer, son système de coordonnées est donné à nouveau. Enfin, si la taille du logo est donnée en pixels, le positionnement de la bulle dans le graphique est donné en "coordonnées géographiques" :-).

Frédéric Mazué - fmazue@programmez.com



Le bug PhotoChooserTask

CONTEXTE

Silverlight pour Windows Phone permet le développement d'application simple et robuste. Pour preuve, lorsque nous voulons accéder à des ressources en dehors de notre application, nous devons nécessairement passer par des classes spécifiques qui se chargeront d'accéder aux parties sensibles du système. On appelle ces classes des *LauncherTask*, si on lance des applications (création d'email, partage de lien sur des réseaux sociaux, ouverture d'internet explorer) ou des *ChooserTask* si on effectue des sélections (contact, adresse email, photo). C'est dans cette dernière catégorie que nous allons traquer un bug susceptible de tuer notre application.

PRÉSENTATION DU BUG

Dans une application Windows Phone développée pour NODO, un bouton appelle un *PhotoChooserTask*, une classe *ChooserTask* chargée de sélectionner une photo. Dans l'émulateur, la sélection d'une photo fonctionne parfaitement. En revanche sur un terminal MANGO, alors que l'application fonctionnait il y a encore quelques mois, notre application crashe parfois sur certaines images sans explication.

ANALYSE

Commençons par analyser le code responsable du désagrément. Le *PhotoChooserTask* est instancié très classiquement :

```
PhotoChooserTask photoChooserTask = new PhotoChooserTask();
photoChooserTask.Completed += new EventHandler<PhotoResult>
(photoChooserTask_Completed);
photoChooserTask.Show();
```

Lorsque la méthode *Show* se termine elle appelle l'évènement *Completed* chargé de générer une image utilisable par Silverlight. C'est certainement ici que l'erreur est générée.

```
void photoChooserTask_Completed(object sender, PhotoResult e) {
    BitmapImage image = new BitmapImage();
    image.SetSource(e.ChosenPhoto);
    this.img.Source = image;
}
```

Comme l'erreur ne se produit pas sur l'émulateur et que Zune empêche tout accès au *PhotoChooserTask* lorsqu'il est branché à notre ordinateur de développement, nous ne pourrions pas déboguer directement dans VisualStudio. Nous sommes donc obligés de recourir à la bonne vieille méthode du *TryCatchMessageBox* :

```
try{
    BitmapImage image = new BitmapImage();
    image.SetSource(e.ChosenPhoto);
    this.Image.Source = image;
}
catch (Exception ex){ // Affichage du nom de fichier et du message d'erreur
    MessageBox.Show("NomdeFichier=" + e.OriginalFileName + " Erreur="
+ ex.Message);};
```

Nous relançons notre application afin de sélectionner une des images source d'erreur. Cette fois-ci l'erreur est attrapée et une boîte d'alerte nous affiche le message suivant (puis l'application se crashe si l'on n'effectue pas d'action sur la boîte) :

```
NomdeFichier=\Applications\(...) \PhotoChooser-aa052fc0-d6ca-4e
42-a249-1b8fc03f72cb.jpg Erreur=The request is not supported.
```

Le message nous apprend que pour une image de format jpeg « La requête n'est pas supportée ». Penchons-nous maintenant plus attentivement sur le répertoire contenant ce fameux fichier. C'est un répertoire SkyDrive, c'est-à-dire relié au Cloud. Ces répertoires sont apparus avec la dernière version de Mango (7.1). En se connectant sur le site web Skydrive, on retrouve notre photo mais celle-ci n'est pas du type escompté : C'est un GIF ! En effet Skydrive supporte des images aux formats JPG, PNG mais aussi GIF, BMP et TIF. Or, Silverlight ne reconnaît nativement que les formats JPG et PNG d'où notre erreur de requête non supportée.

RÉSOLUTION

Pour traiter le problème il suffit de catcher l'erreur comme on l'a déjà fait plus haut. Attention à ne pas utiliser, cette fois, de *MessageBox* dans la méthode *Completed* car elle ne peut excéder, par design, quelques secondes d'activité. On va donc déporter la *MessageBox* dans la méthode *OnNavigated* qui est appelée au lancement de la page.

```
bool haveBadFormatFile = false;

protected override void OnNavigatedTo(System.Windows.Navigation
.NavigationEventArgs e) {
    if (this.haveBadFormatFile == true)
    {
        haveBadFormatFile=false;
        MessageBox.Show("Mauvais format de fichier !");
    }
    base.OnNavigatedTo(e);
}

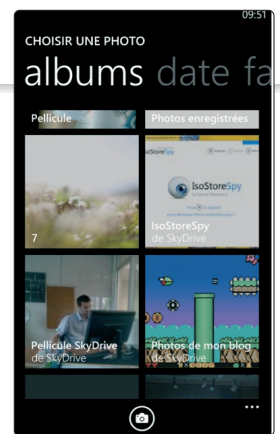
void photoChooserTask_Completed(object sender, PhotoResult e)
{
    this.haveBadFormatFile = false;
    try{
        BitmapImage image = new BitmapImage();
        image.SetSource(e.ChosenPhoto);
        this.Image.Source = image;
    }
    catch{
        this.haveBadFormatFile = true;
    };
}
```

Une enquête rondement menée pour un coupable dans les nuages !

Samuel **Blanchard**

Responsable développement chez Naviso

Blog : <http://blog.naviso.fr> - Site : <http://www.naviso.fr>



Le PhotoChooserTask en pleine action

Les outils des Décideurs Informatiques

*Vous avez besoin d'info
sur des sujets
d'administration,
de sécurité, de progiciel,
de projets ?*

*Accédez directement
à l'information ciblée.*

Cas clients
Actu triée par secteur | **Avis d'Experts**



Actus / **Evénements** | **Newsletter** | **Vidéos**



www.solutions-logiciels.com

☐ **OUI, je m'abonne** (écrire en lettres capitales)

Envoyer par la poste à : Solutions Logiciels, service Diffusion, GLIE - 17 chemin des Boulangers 78926 Yvelines cedex 9 - ou par fax : 01 55 56 70 20
1 an : 50€ au lieu de 60€, prix au numéro (Tarif France métropolitaine) - Autres destinations : CEE et Suisse : 60€ - Algérie, Maroc, Tunisie : 65€ , Canada : 80€ - Dom : 75€ - Tom : 100€
10 numéros par an.

☐ M. ☐ Mme ☐ Mlle Société

Titre : Fonction : ☐ Directeur informatique ☐ Responsable informatique ☐ Chef de projet ☐ Admin ☐ Autre

NOM Prénom

N° rue

Complément

Code postal : Ville

Adresse mail

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

Control Your Cross-Platform Experience

check out infragistics.com/jquery



COMBO

The fully featured combo box control offers intuitive auto-suggest, auto-complete and auto-filtering built in.

HIERARCHICAL GRID

An expandable data grid that presents multiple parent-child relationships is the backbone of your data application.

HTML EDITOR

Give your users a powerful HTML editing experience by incorporating the jQuery WYSIWYG editing tool.

TREE

Simplify the look of hierarchical data, while offering the experience, design and functionality your users will love!

BUSINESS CHARTING

Combine interactive Outlook style grids with rich business charting to deliver a complete portable solution.



Infragistics Ventes France 0800 667 307 • Infragistics Ventes Europe +44 (0) 800 298 9055 • Infragistics India +91 80 4151 8042 • [@infragistics](https://twitter.com/infragistics)

Copyright 1996-2011 Infragistics, Inc. All rights reserved. Infragistics and NetAdvantage are registered trademarks of Infragistics, Inc. The Infragistics logo is a trademark of Infragistics, Inc.

