

C - C++ - JAVA - VISUAL BASIC - DELPHI - XML - VBSCRIPT

Programmez!



# Programmez!

39 F

AVRIL 1999 • N° 9

le 1<sup>er</sup> magazine de tous les langages

# C++

## Nuit et jour !

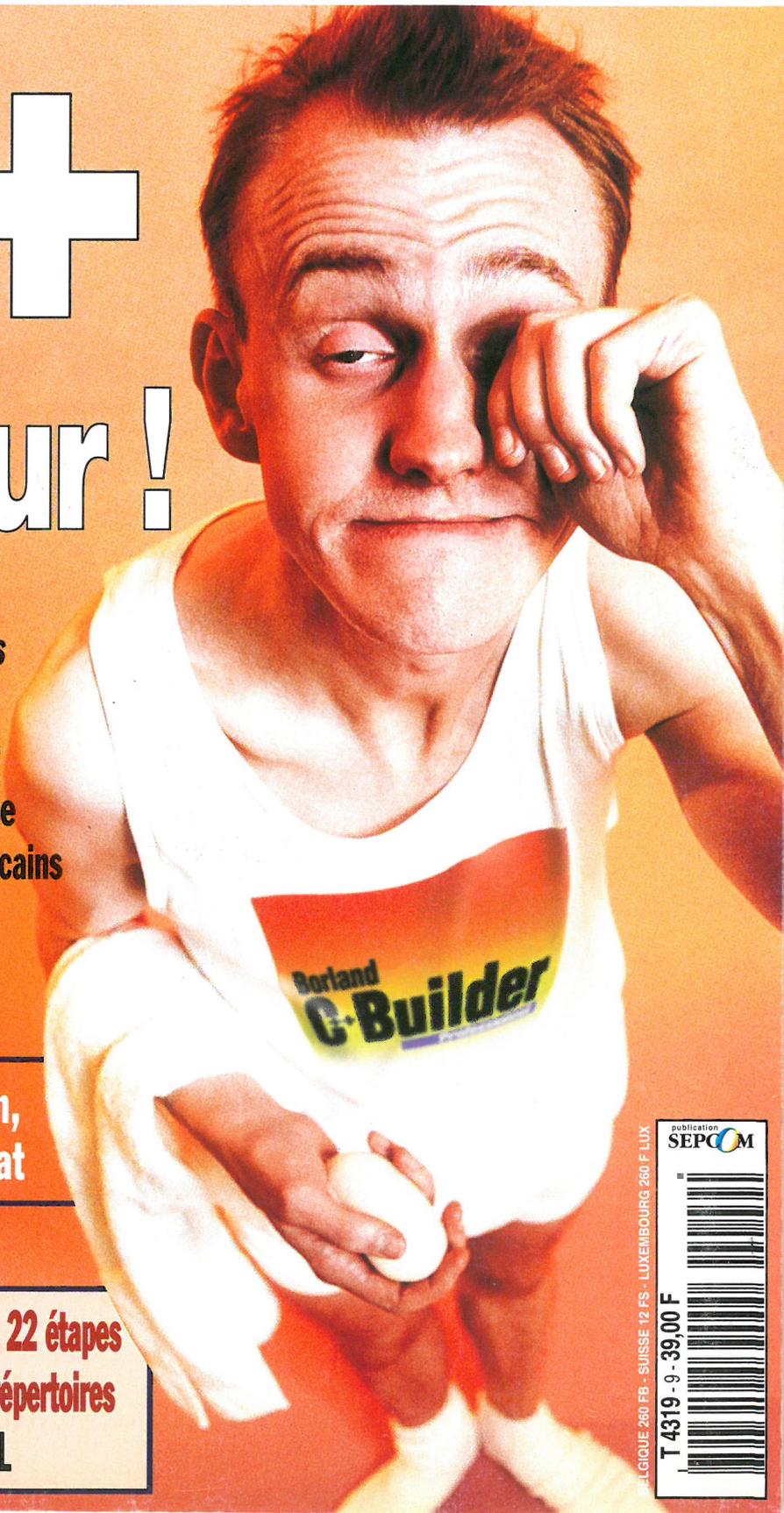
- **INITIATION** la vie des objets
- **COMPILATION** concevoir des *makefiles*
- **MEMOIRE** les doubles listes chaînées
- **SYSTEME** la programmation du clavier
- **TEST** le dernier C++ Builder 4 d'Inprise
- **RECRUTEMENT** les exigences des Américains
- **EXPERT** les secrets du signe égal

### ACTUEL

Bases de données **XML**, législation, gestion de projet, logiciels de l'Etat

### PRATIQUE

**WORKFLOW** faites circuler l'information en 22 étapes  
**VISUAL BASIC** imprimez le contenu de vos répertoires  
**MULTIMEDIA** astuces pour Java 3D et VRML



BELGIQUE 260 FB - SUISSE 12 FS - LUXEMBOURG 260 F LUX

publication SEPCOM



T 4319 - 9 - 39,00 F

**Société Européenne de Presse et de Communication**  
5-7 rue de l'Amiral-Courbet,  
94160 Saint-Mandé - France  
Tél. : 01 43 98 22 22 - Télécopie : 01 43 98 43 82

**Editeur délégué :**  
Lionel Pillet

**Rédaction**  
Directeur : Michel Barreau  
Rédacteur en chef :  
Matthieu Thfoin ([feedback@planetepc.fr](mailto:feedback@planetepc.fr))  
en collaboration avec Anne Vincensini  
(secrétaire de rédaction)

**Ont collaboré à ce numéro...**  
Jérôme Arnaud, Alain Baritault, Nicolas  
et Jean-Bernard Boichat, Claude Catonio,  
Sébastien Marc, Frédéric Mazué, Marc  
Perroulaz, Jean-Paul Pruniaux, Frédéric Reitz,  
Henri Ruch, Rémi Sussan (rubrique multimédia),  
François Tonic, Sylvie Zabroniecka  
Remerciements à Stéfane Fermigier (AFUL)  
et Arnel Guillet (CIGREF)

**Maquette**  
Directeur artistique, Xavier Chambon,  
en collaboration avec Véronique Belpois  
et Valérie Guihal

**Services en ligne (Isicom)**  
Directeur : Stéphane Kaminsky ([sk@isicom.fr](mailto:sk@isicom.fr))

**CD-ROM**  
Chef de produit multimédia :  
Olivier Delacour ([delacour@isicom.fr](mailto:delacour@isicom.fr))  
Développement multimédia :  
Philippe Coupez ([pcoupez@isicom.fr](mailto:pcoupez@isicom.fr))  
Responsable shareware/Hot-line :  
René Robelin ([hotline@isicom.fr](mailto:hotline@isicom.fr))

**Publicité**  
Directrice commerciale de la publicité :  
Sylvie Hugonnier (43 86)  
Chef de publicité SEPCOM :  
Bernard Bibas (43 65),  
Régie publicitaire YellowBee : Fabien Carletto  
Tél. : 01 39 38 23 41  
E-mail : [fcarletto@yellowbee.com](mailto:fcarletto@yellowbee.com)  
Assistante de publicité : Christine Gaugry (43 83)

**Diffusion**  
Directeur : Thierry Cagnion  
Assisté de Vanessa Pénélaud (88 43)  
Sepcom Service abonnements :  
78 rue La Condamine, 75017 Paris.  
Tél. : 01 44 90 70 55,  
du lundi au vendredi  
de 9h30 à 12h30 et de 13h30 à 17 h.

**Fabrication**  
Daniel Nardon

**Directeur de la publication :**  
Jean Kaminsky  
PDG, principal actionnaire

**Groupe Sepcom**  
SA au capital de 450 000 F

**Impression**  
Imprimerie Brodard graphique  
Numéro de commission paritaire : 0900 K 78366  
Dépôt légal : 2<sup>e</sup> trimestre 1999

Ce numéro comporte un encart "Abonnement"  
broché entre les pages 34 et 35, un encart "Sun"  
jeté au dos de la 4<sup>e</sup> de couverture sous blister  
et un encart "Sybase" jeté.

Groupe  
**SEPCOM**

**HOT LINE CD-ROM**  
par Minitel : 3614 ISICOM  
par Internet : [cdrom@planetepc.fr](mailto:cdrom@planetepc.fr)  
par téléphone : 01 43 98 43 89  
du lundi au vendredi

## C++, Web --

**N**uits et jours, des programmeuses et des programmeurs utilisent le C++, dans le rush des projets informatiques. De Wall Street à Bruxelles en passant par Poissy, le C++ est le pilier du développement, aussi bien dans les usines à code, les départements de R&D ou dans le coin bureau du domicile. Les plus grands logiciels, best-sellers de tous les domaines, sont générés dans ce langage. Non, le C++ n'est pas une troisième langue pour celui qui l'apprend, mais il sert de point commun à l'industrie mondiale du développement.

Il est plus important de bien causer C++ que de parler un américain sans accent si on souhaite se faire embaucher à New York. Etre ou ne pas être standard, être beau, compact et élégant, on ne sait pas ce que les listings pensent quand ils se regardent dans la glace.

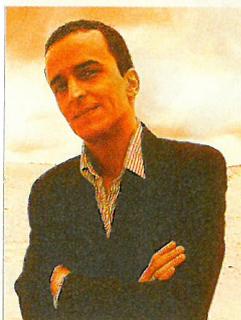
Les professionnels et particuliers ont choisi le C++ pour sa puissance et ses possibilités multiples, qu'elles soient objet ou non. A ces critères objectifs s'ajoute un coefficient passionnel non négligeable. Le C++ reste un langage très affûté, même pour CORBA

et DCOM, grâce à la nouvelle génération d'outils RAD, comme le tout dernier C++ Builder ou les outils de gestion intégrale de projet, tel celui de Continuous.

Indépendamment de ce constat objectif sur un langage-clé, l'actualité du développement est un peu perturbée par deux mouvements juridiques contradictoires, préoccupants pour ceux qui s'intéressent au monde réel.

Les législations de propriété intellectuelle et de services télématiques sont-elles entre de bonnes mains ? Bientôt il faudra payer des royalties pour penser et avoir vingt-cinq avocats par site hébergé sur le web.

D'un côté les grandes entreprises et, par extension, les PME/PMI demandent un renforcement de la législation Internet pour protéger les noms de domaine et favoriser le commerce électronique. Une exigence justi-



fiée par de nombreux abus et un certain impérialisme exercé sur le Web, qui menace l'économie nationale et européenne.

De l'autre côté, les fournisseurs de services comme l'hébergement gratuit n'ont pas les moyens de contrôler tous les contenus (voir [www.altern.org](http://www.altern.org)). Ainsi, un indélicat a publié des

photos d'une célébrité nue. La condamnation du fournisseur AlternB entraîne la fermeture de plus de 47 000 sites. La condamnation fait jurisprudence et inspire les hautes sphères du pouvoir européen.

La Commission européenne, temple du lobbying, semble "favoriser" une législation informatique et télématique qui coupe l'herbe sous le pied du logiciel libre, de la philosophie humaniste et pénalise l'internet non-lucrative. A long terme, si on développe les brevets informatiques, ce seront les PME de développe-

ment qui seront écrasées par les géants du logiciel, qui sont mieux armés financièrement.

Comment concilier les justes droits tels que faire du commerce électronique avec un nom

correct pour son site, faire respecter les lois et,

par ailleurs, la possibilité d'apporter une brique personnelle à l'œuvre universelle du Web, avec un esprit ludique, expérimental, inventer des métiers et des modèles économiques originaux, sans que des technocrates et des censeurs illégitimes aient quelque chose à redire. Internet n'appartient pas aux entreprises, surtout aux pieuvres, ni aux associations, surtout de malfaiteurs, ni aux particuliers, surtout les cas. Si Internet n'est pas un espace de civilisation, de consensus, de pluralité et de citoyenneté, il faut s'en déconnecter tout de suite. On ne fait pas de business sur un marché pareil. ■

*Matthieu Thfoin*

**Matthieu Thfoin, Rédacteur en Chef**  
[feedback@planetepc.fr](mailto:feedback@planetepc.fr)



## CLES DU SOFT

### 6 Actualités

**Continuus** invente la gestion de projet intégrale, **XML** et les nouveaux serveurs d'information, le **Cigref** défend les sites des entreprises, **Sun** s'ouvre aux portails, les succès-services d'**IBM**, l'**Aful** dénonce la législation des brevets et les sanctions contre les hébergeurs.

### 13 Interviews

**Dominique Goupil**, le président de **FileMaker**, et **Godefroy Beauvallet**, chargé de mission du Premier Ministre dans le cadre des **MTIC**.

### 16 Utilitaires

**BeOS 4**, le plus beau, **LinuxPPC**, un Linux pour Power Macintosh et l'atelier de génie logiciel **Windev**.

## LANGAGES D'AVENIR

### 24 Delphi

Accéder aux tables complexes du **SGBD Paradox**

### 28 WorkFlow

Comment mieux faire circuler l'information avec l'outil **PeopleSoft** ? 22 étapes suffiront.

### 38 C++ l'opérateur égal

Un projet leader de Wall Street raconte les méthodes de recrutement et livre les secrets de l'opérateur égal.

### 46 C++ Liste

Tout ce que vous avez toujours voulu savoir sur les doubles listes chaînées génériques.

### 48 C++ Hook

Le crochet attrapant les messages du système et les remplaçant par les vôtres. Exemple : le clavier

### 54 Apple Guide

Générer des aides pour Macintosh

### 81 C++ le jeu de la vie

Initiation à la programmation objet

### 62 VBScript

Maîtriser les nouveaux scripts pour Windows et DOS.

### 64 C, mon make à moi

La star des GNU pour une compilation optimale

### 68 VB, impression répertoires

Comment lister et imprimer les fichiers stockés.

## BONUS

### 89 Multimédia

Java 3D + VRML 97

### 94 Livres

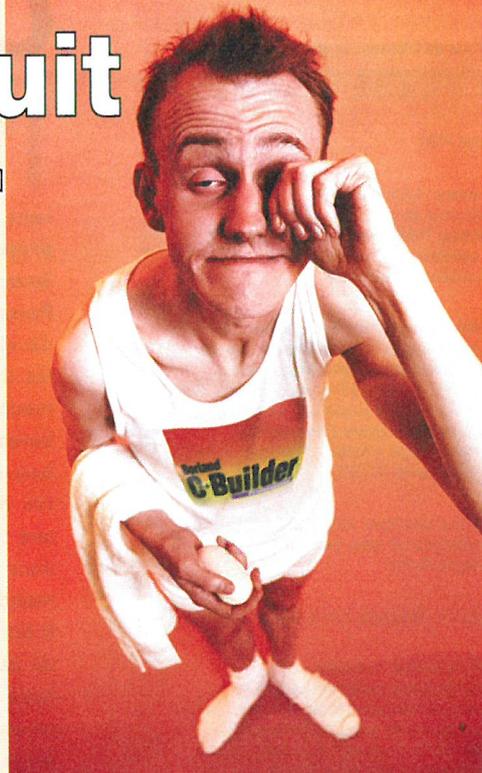
# > Sommaire

## EN COUVERTURE

# C++, nuit et jour

- > Opérateur égal
- > Recrutement à l'américaine
- > Hook du clavier
- > Listes chaînées
- > La vie en objet
- > Makefiles optimisés.

p.38



## Langages

# Delphi, Visual Basic VBScript

Les solutions pratiques

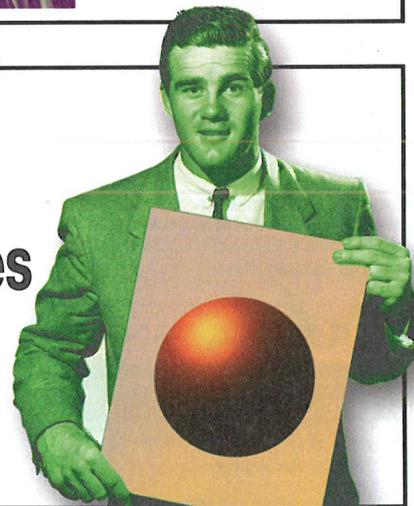
p.24

## Multimédia

# Siècle des lumières virtuelles

Les astuces d'éclairage en Java3D et VRML...

p.89

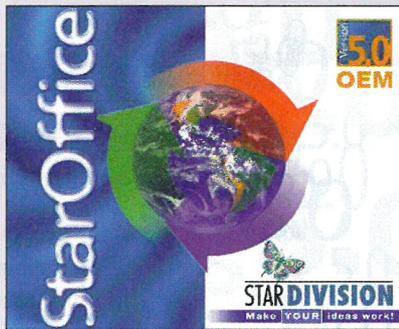


PROGRAMMES

Star Office 5.0 pour LINUX :

(version complète et illimitée après enregistrement)

Voici la version 5.0 de la célèbre suite bureautique alternative. Elle comprend différents modules :



- un module traitement de texte
- un module tableur
- un module dessin
- un module présentation
- un module base de données
- un module accès internet

Important :

1/ Afin d'installer ce logiciel, il est indispensable de saisir le code personnel qui vous est indiqué au dos de la jaquette du CD-ROM, sur une étiquette autocollante.

2/ Cette version complète est librement utilisable pendant trente jours suivant son installation à la suite de quoi, il vous faudra vous enregistrer : à l'ouverture du logiciel, cliquer sur " Oui " dans la boîte de dialogue " Star Office 5.0 OEM " et ensuite, lorsque la boîte de dialogue " Inscription " apparaît, cliquer sur " S'inscrire " et suivre les instructions

Sun JDK 1.2 :

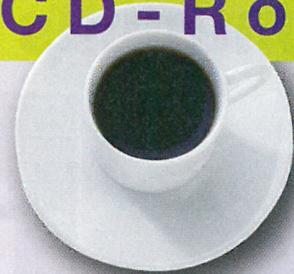
(kit complet de développement Java)

Cette version complète 1.2 du JDK (Java Development Kit) pour Windows dispose des nouvelles fonctionnalités de la plateforme Java 2.

Offert par Sun Soft.



sommaire CD-Rom



Tango Development Studio de PERSVASIVE

(version complète et illimitée après enregistrement)

Cette version promotionnelle vous permet d'installer le studio de développement Tango qui comprend différents modules.



Important :

Afin de bénéficier de votre version complète de Tango Development Studio, il est nécessaire de vous enregistrer sur le site web de Pervasive à l'adresse suivante : [www.register.pervasive.com](http://www.register.pervasive.com).

Cette version n'est pas une version de démonstration mais bien une version complète de l'outil de développement Tango.

Vous pouvez également envoyer vos coordonnées par courrier ou fax chez Pervasive France, Immeuble

Atria, 21 Avenue Edouard Belin, 92 500 Rueil Malmaison fax: 01 55 47 17 07.

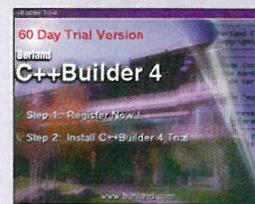
N'oubliez pas de Télécharger votre version complète de Pervasive.Tango à l'adresse suivante

[www.pervasive.com/products/tango/webjump/](http://www.pervasive.com/products/tango/webjump/)

Gnat Ada :

Langage complet à la norme ADA 95 en licence libre.

Borland C++Builder 4.0



Cette version d'évaluation de 60 jours vous permettra de tester la nouvelle

version 4.0 de Borland C++Builder.

Avant l'installation, veuillez lire les fichiers suivants : install.txt, readme.txt, license.txt, register.txt. (présents dans le répertoire \logiciels\bcpp.

Erratum :

La version d'évaluation du logiciel C++ Builder 4 de Borland est à nouveau présente sur le CD-ROM de votre magazine.

En effet, à la suite d'une malencontreuse erreur technique de notre service multimedia, la version présente le mois dernier n'était pas totalement utilisable.

Aussi, nous tenons à adresser à la société INPRISE ainsi qu'à nos nombreux lecteurs, toutes nos excuses pour ce désagrément et espérons que vous aurez d'autant plus de plaisir à découvrir ce fabuleux produit, ce mois-ci.

Tron :

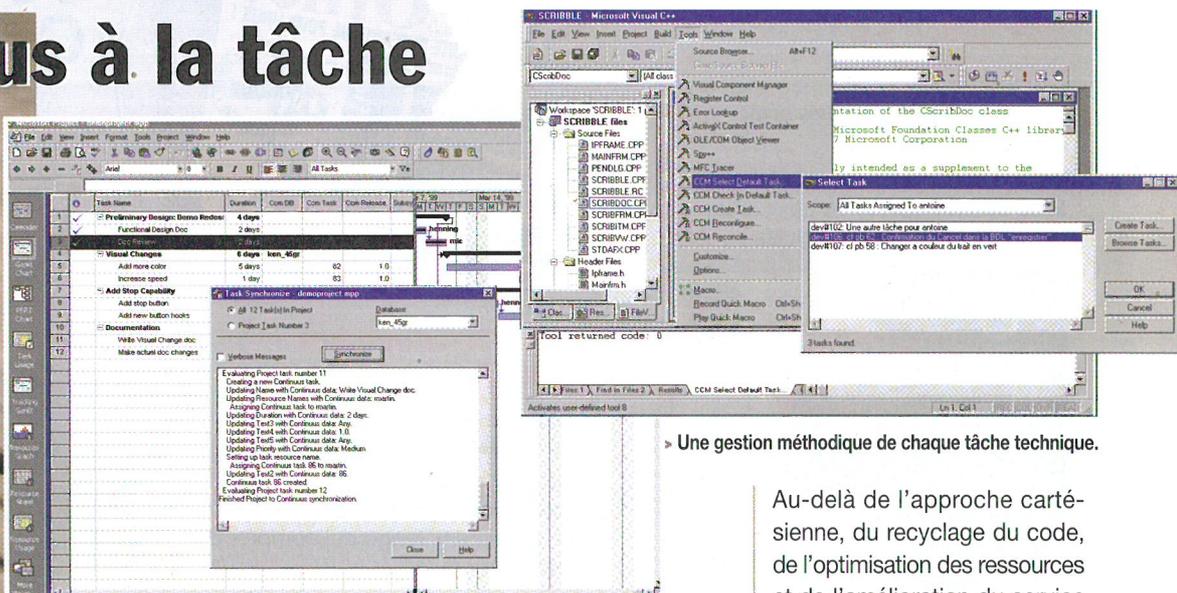
Les ressources libres du projet leader de l'industrie et des applications embarquées au Japon.

LISTING DES ARTICLES

Retrouvez les sources des articles dans le dossier REDAC à partir de la racine du CD.

## PROJET

## Continuus à la tâche



► Une gestion méthodique de chaque tâche technique.

► Les informations du projet remontent dans le logiciel de planning.

**S**pécialiste de la gestion de projet de haut de gamme, l'éditeur américain ne travaille que sur les applications critiques pour les entreprises. Le fait d'être orienté process distingue ce produit, qui est une sorte de portail technique servant aussi bien de bureau des développeurs pour l'utilisateur que de centre nerveux pour les créateurs de logiciels.

La solution CM/Suite répond à une problématique des Project Leader : comment relier vos gestionnaires de configuration, de traitement d'erreur et outils de planning sans aucune ligne de script. Cette mission est impossible avec les outils classiques, imposant un minimum de développement. Il y a même souvent une phase purement manuelle pour mettre à jour la gestion de projet. Continuus est le seul à raccorder les équipes de développement avec les bêta testeurs, les clients, le support technique, sans script, et le chef de projet bénéficiera de mises à jour dynamiques dans Microsoft Project, grâce au modèle d'intégration COM.

D'après Jean-Louis Vignaud, le directeur technique de Continuus en France, seulement 16 % des projets de développement informatique aboutissent et 50 % coûtent deux fois plus cher, 31 % sont arrêtés et 50 % des directeurs informatiques pensent qu'il y a plus d'échecs qu'auparavant.

## 16 % de réussite

L'adoption d'une telle solution nécessite une prise en charge par des consultants pour adapter les méthodes de travail et les options de configuration. La partie cliente WebSynergy propose une interface DHTML pour la saisie des fiches d'anomalie, le helpdesk et le suivi du cycle de vie des demandes. Cet accès par le Web facilitera également le reporting à distance. Continuus va gommer petit à petit les soucis du chef de projet. Vous aurez des traces des demandes de client et des certitudes sur leur satisfaction. La gestion de configuration automatisée et l'encadrement des modifications de code appar-

teront une sérieuse économie en temps et donc en ressource. Enfin, l'intégration dans les logiciels de planning assure une gestion plus réaliste des planning, c'est-à-dire en fonction de l'avancement réellement enregistré par le système CM. Continuus réduit aussi le fardeau des rassembleurs de projet. Finis le stress du dernier moment et les problèmes de livraison, la perte de certains travaux, les modifications sur la version de développement et non sur celle en exploitation, ou encore le casse-tête pour annuler une implémentation.

## Adieu casse-tête

Enfin, côté développeur, la suite CM de Continuus leur épargnera des conflits de versions, la perte de fichiers, les conflits d'accès aux fichiers partagés, ainsi que la difficulté de prise en main des projets développés par d'autres équipes. Mais l'outil de Continuus sera clairement un Big Boss, l'œil du chef de projet dans le cambouis du programmeur.

Au-delà de l'approche cartésienne, du recyclage du code, de l'optimisation des ressources et de l'amélioration du service client, ce produit permet de bâtir un référentiel technique et commercial, véritable trésor de guerre de l'entreprise. Il s'intègre avec les outils de développement les plus utilisés (Visual Studio, PowerBuilder, Oracle Developer 2000, Uniface, Visual Café, C++ Builder, Delphi et d'autres à venir). Selon l'éditeur, les entreprises remboursent leur investissement en licence au bout de 3 à 6 mois d'activité.

Le serveur pour Windows NT coûte 100 000 francs environ et comptez 50 000 francs de plus sous Unix. La licence client est de 16 000 francs avec un tarif très dégressif. Cette solution s'adresse donc aux sociétés comptant suffisamment de développeurs pour mieux amortir le serveur. ■

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

## ...EN BREF

### WorkFlow

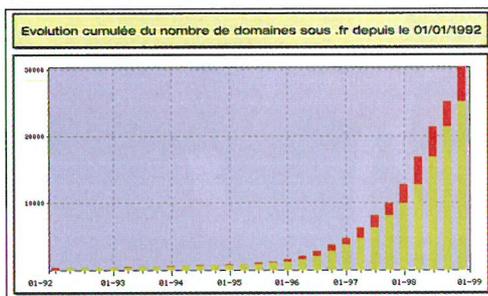
● CSE Systems, spécialiste du WorkFlow, déjà implanté en Europe, s'installe en France.

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

## Nom d'un site

# Le Cigref contre-attaque

**S**elon le Club informatique des grandes entreprises françaises (Cigref), une entreprise sur deux rencontre des problèmes avec les noms de domaine Internet, c'est-à-dire le nom des sites [www.dupond.com](http://www.dupond.com) (déjà réservé, déposé exprès pour la vente ou presque identique à l'original). Ces problèmes sont des handicaps pour le développement de l'Internet en France et en Europe.



Les noms de domaines français (.fr) sont gérés par le conseil d'administration de l'AFNIC, dont le Cigref fait partie. Aux Etats-Unis, l'organisme de référence pour l'évolution des noms de domaines est l'Icann (Internet Assigned Corporation for Names and Numbers). Selon le Cigref, les sites commerciaux (.com) posent beaucoup plus de problèmes. L'objectif des représentants français est de réduire le plus possible les zones de non-droit sur Internet, qu'il s'agisse de celui des nations, des entreprises ou des particuliers. Plus d'universalité et d'arbitrage sont attendus à l'avenir. Les dix recommandations du Cigref se résument en substance à :

1- Déposer votre société dans

> Un nombre de domaines français en forte hausse depuis 1997.

les domaines .fr et .com à titre conservatoire.

2 - Réunir tous les services de l'entreprise pour établir une stratégie.

3 - Définir les marques, dénominations, raisons sociales, enseignes, noms de projets... qui imposent des dépôts dans tous les pays de l'activité.

4- Toujours payer les cotisations annuelles pour les dépôts.  
5 - Se faire référencer par tous les carrefours d'internautes (moteurs de recherche, annuaires, portails).

6 - Demander l'assistance des organismes représentatifs (chambre syndicale, chambre de commerce, associations

professionnelles...) en cas de problème.

7 - Recourir si nécessaire à des compétences externes (veille technologie et économique, assistance juridique).

8 - Adopter une attitude juridique musclée face aux abus et communiquer sur les actes de malveillance dont l'entreprise est victime.

9 - Analyser la fréquentation des sites selon le nom de domaine utilisé pour mieux ajuster votre stratégie.

10 - Chaque nouveau nom inventé dans l'entreprise doit être protégé dans le cadre d'Internet. ■

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

**AFNIC**  
Association Française pour le Nommage Internet en Coopération

Venez adhérer à l'AFNIC! | Initiez vous à l'Internet!

- STATISTIQUES**: Statistiques sur l'internet en France, en Europe, dans le monde. La charge de nos DNS.
- PRESTATAIRES**: La liste des prestataires Internet, les concessionnaires AFNIC et les non-concessionnaires.
- ACTUALITES**: Les nouveautés, Les manifestations, Les résultats de la consultation.
- GUIDES**: Les Guides sur l'internet, le F.A.Q. notre serveur F.T.E. et nos formations.
- INFORMATIONS**: Le statut de l'AFNIC, La C.A. de l'AFNIC, Son équipe opérationnelle, Comment nous contacter?
- RECHERCHES**: L'annuaire de la zone .fr, Recherche d'info via D'Info.
- OUTILS**: ZoneSearch, TraceTools, FindParent, Interrogation du DNS pour la France, Le logiciel Bind.
- ENGLISH VERSION**: English version.
- PROCEDURES**: Les procédures, Le formulaire en ligne et la lettre d'engagement, La Consultation et le Charte de nommage.

> L'AFNIC gère les noms de domaines .fr.

## ...EN BREF

### Supports

● Le Syndicat national des supports d'enregistrement annonce trois hausses spectaculaires entre 97 et 98. Les Minidiscs augmentent de 288 %, quand les MiniDV jaillissent à 342 % et les CD-R s'écoulent à 250 %. Coté piratage, il vaut mieux vendre le contenant que le contenu.

### Projection

● Le Forum des arts de l'univers scientifique et technologique (Faust) s'est déroulé en octobre dernier à Toulouse. La projection des œuvres primées (CD-ROM, sites Internet et films) aura lieu les 22 avril, 27 mai, 23 septembre et 21 octobre au centre multimédia de l'ADAC, 13 place Nationale dans le 13<sup>e</sup> arrondissement de Paris.

## GESTION COMMERCIALE

# Aussi sur Linux

**B**ARTH Group's va porter sous Linux une importante application de gestion commerciale, GESTECO. C'est une application de gestion commerciale multi-points de ventes, permettant la gestion des différents domaines comme les achats, les stocks, les ventes et les clients. Ecrite en C ansi, elle est actuellement en discussion pour être réécrite en C++ et portée dans une version

générique sous le système d'exploitation Linux. Cette application, commercialisée à plus de 600 exemplaires dont beaucoup en France dans la parfumerie sélective, va donc être proposée dans l'environnement Linux, apportant à celui-ci un apport de sang neuf dans les applications commerciales, ce qui manque le plus à ce système. ■

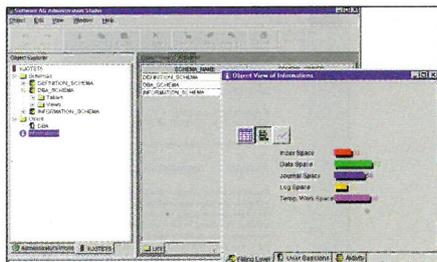
[www.users.skynet.be/structure](http://www.users.skynet.be/structure)

Telex : ..... Centura Software lance  
SQLBase, aux informations cryptées.....

## XML

# Plus de serveurs

Comme nous vous l'annonçons dans notre numéro de novembre, XML n'est pas seulement un successeur ou une alternative au HTML. C'est un format plus important qu'il n'y paraît, dans la mesure où il aura un impact sur toute l'informatique interne. Le client XML n'est pas un souci en soit, car presque tous les navigateurs s'y mettent et certainement les logiciels bureautiques. C'est plutôt du côté serveur que tout se joue. On ne compte plus les sociétés



> Le management du serveur Tamino.

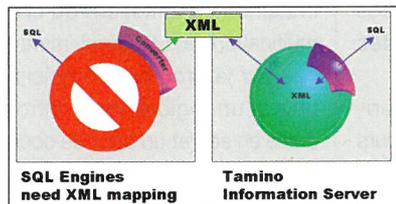
un serveur Web et un SGBD, doté de capacités transactionnelles pour les applications internes et aussi le commerce électronique.

Les autres serveurs reprennent les mêmes principes, mais avec des positionnements différents. Interleaf et ObjectDesign s'ar-

rangent sur les licences de leurs produits respectifs, un compositeur de pages et un gestionnaire de contenu, qui s'avèrent complémentaires. Ardent Software, fournisseur de SGBD objet et d'outil de Data Warehousing,

voit XML comme un format d'échange de données, permettant d'extraire de tous les systèmes (ERP, base de données) les informations, pour mieux les exploiter. Actuate est soutenu par PeopleSoft, Ariba et Netscape pour que cette société développe son outil XML Reports, fournissant à terme des rapports personnalisés sur l'activité des sites de commerce électronique. Enfin, Softeam s'inspire de l'XML en soutenant l'XMI, un format d'échange entre les AGL et les outils de modélisation UML. ■

[www.w3c.org](http://www.w3c.org)  
[www.softwareag.com](http://www.softwareag.com)  
[www.ardentsoftware.com](http://www.ardentsoftware.com)  
[www.actuate.com](http://www.actuate.com)  
[www.odi.com](http://www.odi.com)  
[www.softeam.fr](http://www.softeam.fr)



> Avec un format natif XML, les performances sont bien meilleures.

majeures qui s'investissent dans XML, chacune à leur façon. Software AG, en partenariat avec DataChannel, propose le serveur d'information Tamino.

## Un format natif XML

L'éditeur allemand a tranché entre deux solutions, un serveur relationnel étendu ou un format natif XML. Et c'est bien cette dernière qui sera retenue, pour des questions de performance. Le support natif n'empêche pas de stocker des données relationnelles dans les champs XML et de les récupérer en exécutant des applications par l'interface SQL de Tamino. Le serveur d'information est à la fois



e-business

## Eléments logiciels pour la construction d'applications de libre-service sur le Web.



**Un fait :** selon l'analyste Jeff Rumberg, de Meta Group, une transaction commerciale coûte en moyenne 30 francs quand elle est faite par un vendeur, 3 francs par l'intermédiaire d'un serveur vocal et quelques centimes avec le Web (InfoWeek 1<sup>er</sup> juin 98).

**Des applications qui travaillent pour vous.** *WebSphere Application Server* combine un environnement d'exécution de servlets Java avec des connecteurs aux bases de données et aux environnements objets. Vous pouvez ainsi développer sur votre site Web des applications critiques à l'entreprise. *WebSphere Studio* vous aide à réaliser cela en un temps record. *WebSphere* inclut notamment *VisualAge for Java, Version 2.0* qui a obtenu le **Prix d'Excellence Technique 1999** décerné par *PC Expert France*.

**Le libre-service en action.** *Lotus Domino Application Server* vous permet de construire des applications collaboratives pour répondre à l'évolution rapide des modèles d'entreprise. Et *Domino* supporte les solutions, sécurisées et hautement interactives, de libre-service sur le Web.

**Avis à la population.** *DB2 Universal Database* a été choisi par certains des sites Web les plus fréquentés pour sa capacité à offrir des performances exceptionnelles à un très grand nombre d'utilisateurs. Entièrement compatible Java, il tourne en natif sur toutes les principales plates-formes.

**L'intégration transparente.** *MQSeries* vous aide à intégrer vos applications nouvelles et anciennes au cœur de vos solutions de libre-service sur le Web. Désormais totalement compatible Java, *MQSeries* est le leader mondial des produits d'intégration et tourne sur plus de 25 plates-formes. Il a en outre reçu le label *Yphise* en décembre 1998.

## Essayez avant d'acheter.

GRATUIT : le kit de découverte  
des solutions IBM de libre-service sur le Web.  
Tout le nécessaire pour commencer à bâtir vos applications  
de libre-service, code d'essai inclus, vous attend sur  
[www.software.ibm.com/webselselfserv/fr](http://www.software.ibm.com/webselselfserv/fr)

## ...EN BREF

## Intranet

● Pour convertir les PME/PMI, Sun, Lotus, Sekoya et Ilion créent le Pack Intranet Business, une sorte d'Intranet clé en main. Disponible en trois choix de 10, 25 et 50 licences, coûtant respectivement 44, 65, et 117 KF, il comprend la machine du serveur, Notes, Domino, un système Intra/Extranet plus un petit crédit d'heures au support technique.

## Télécommunications

● Le langage ODL (Object Definition Language) est promu par l'Union Internationale des Communications. Il devrait palier aux carences de l'IDL (Interface Definition Language) actuellement supporté par l'OMG. Ce dernier est plutôt conçu pour l'informatique, alors que le nouvel ODL se tourne vers le secteur des télécommunications. Il permettrait toutefois de concevoir des analyseurs syntaxiques, des générateurs de code, les éditeurs de spécifications et les outils associés.  
[www.iut.int](http://www.iut.int)

## Serveurs

● BMC Software porte son outil d'administration Patrol sur le système Linux, pour profiter du bon tiers des serveurs Web en licence libre.

## LÉGISLATION

## Les lois informatiques à la dérive ?

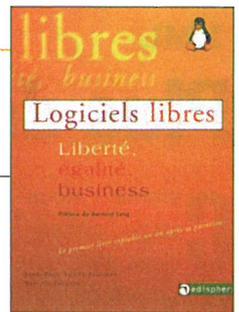
L'AFUL (Association française des utilisateurs de logiciels libres) est contre la responsabilisation juridique des diffuseurs d'information. La justice française vient en effet de faire porter la responsabilité d'une publication illégale, attentant à la vie privée d'autrui, à l'hébergeur d'un site et non à l'auteur de la publication, alors que des moyens techniques auraient permis de déterminer l'identité de l'auteur en infraction et de le poursuivre.

Elle agit conformément à une tendance inquiétante du droit de la propriété intellectuelle qui tend à assimiler les diffuseurs (fournisseurs d'hébergement ou de connectivité) à des directeurs de publication :

- Le Parlement européen a récemment voté en faveur d'une application stricte du droit d'auteur au contenu des caches Web, un dispositif technique utilisé par des opérateurs de télécommunication comme France Telecom pour optimiser les performances du réseau Internet.
- Aux Etats-Unis, le "Digital Millenium Act" permet de couper automatiquement les publications sur le Web soupçonnées d'illégalité, sans même un procès.
- Pour imiter la loi américaine sur les brevets, la Commission européenne (ou ce qu'il en reste) envisage d'autoriser les brevets sur les logiciels et de rompre avec la tradition humaniste européenne, qui voulait que l'on ne s'approprie pas les concepts ou les idées. En



> Le site de la Commission européenne et l'ouvrage qui la contredit.



outre, au lieu de rendre les utilisateurs du brevet responsables en cas de contrefaçon, la proposition de directive européenne propose, comme aux Etats-Unis, de rendre les diffuseurs responsables des contrefaçons.

Or la responsabilisation juridique généralisée des diffuseurs d'information présente plusieurs inconvénients majeurs :

- Elle favorise les grandes entreprises qui ont les moyens financiers d'héberger un site et de se défendre en cas d'attaque. Elle freine l'accès des autres à la publication sur le Web. Quelle société d'hébergement ou quel opérateur de télécommunication accepterait en effet de véhiculer des données au contenu a priori incertain alors que de grandes entreprises proposent un contenu dont elle garantissent la légalité ? En ce sens, la responsabilisation des diffuseurs contribue à promouvoir une société où toute l'information vient de quelques grands acteurs institutionnels ou commerciaux.
- Elle touche de plein fouet les logiciels libres qui sont caractérisés par un modèle de développement où n'importe qui

peut participer. C'est l'une des forces du modèle car les erreurs sont corrigées rapidement et les logiciels évoluent en fonction des besoins des utilisateurs. Imaginons maintenant qu'une multinationale du logiciel, menacée par la progression irrésistible d'un logiciel libre, introduise en secret un bout de code breveté dans ce logiciel. L'hébergeur qui le propose en téléchargement risquerait alors de se faire attaquer pour contrefaçon. Ce scénario n'a hélas rien d'incongru : Microsoft, en tant que membre du World Wide Web consortium (W3C), a déjà introduit en secret des technologies brevetées dans la norme HTML (voir [www.free-patents.org](http://www.free-patents.org)).

La démocratie commence par le respect des lois. Mais certaines lois peuvent aussi tuer la démocratie en altérant l'égalité des citoyens face à l'information, en promouvant une société où les producteurs de logiciels et d'information sont uniquement des grandes entreprises et où les individus peuvent se trouver pénalement responsables lorsqu'ils produisent des logiciels et de l'information utiles à tous. ■

Telex : ..... DirectX 6.1 disponible  
avec son nouveau SDK contenant DirectMusic.....

SSII

## IBM Global Services fait un tabac

IBM Global Services accompagne la transformation des entreprises. Réorganisant les anciens métiers, cette société de service préfère miser sur des compétences stratégiques. Acteurs de cette mutation, les "e-people" sont les consultants du prochain millénaire...

La mondialisation des relations d'affaires encourage l'essor des prestations de services informatiques. En 1997, IBM France a regroupé l'ensemble de ses filiales de services (telles Axone, CGI, Cimad France, Eurequip, IBM Intégration de Systèmes, Intexis, LY-D Informatique, Seprim) sous la marque IBM Global Services.

### La France en poupe

Depuis, la France fait figure de modèle puisqu'elle bénéficie d'un taux de croissance de 20 %, conforme à la moyenne mondiale et à la progression française. Le chiffre d'affaires mondial enregistré en 1998 s'élève à 23,4 milliards de dollars (hors maintenance, celle-ci représentant six milliards de dollars), plaçant IBM Glo-



> L'antenne française de la SSII est florissante.

bal Services au rang de leader mondial des services informatiques.

Près de 40 milliards de dollars de contrat ont été signés en 1998, dont la moitié en infogérance (Strategic Outsourcing). Dans cette catégorie d'activité, IGS a signé en France des contrats avec Rhône-Poulenc, la Lyonnaise des Eaux, la Banque Worms, Hertz, la Fnac... L'infogérance représente donc actuellement l'un des axes majeurs du développement d'IGS, parallèlement à

cinq autres types de services qui permettent d'accroître la réactivité et la compétitivité de l'entreprise : l'e-business (commerce électronique), l'Enterprise Resources Planning (optimisation des processus entreprise), le Customer Relationship Management (maîtrise de la relation client), le Supply Chain (l'optimisation de la chaîne logistique). Tous ces champs d'application correspondent à ce qu'il est aujourd'hui convenu d'appeler les "technologies émergentes". ■

### SERVICES

## Près de 200 milliards de francs en 2002

Si, en France, le taux de croissance des services en 1998 par rapport à 1997 s'est élevé à près de 18 %, il continuera de croître en 1999 à un taux avoisinant les 15 %. Entre 1997 et 2002, selon IDC, ce marché français doublera pour s'approcher de la barre des 200 milliards de francs. Parmi les technologies émergentes, la plus forte croissance reviendra sans nul doute au com-

merce électronique : de 28,20 millions de francs, en 1997, il s'établira à près de 1,5 milliard de francs en 2002, suivant un taux de croissance annuel moyen de 120,5 %. Au second rang des croissances se trouvent les Customer Cares, qui devraient connaître un taux moyen de 68 %, alors qu'ils avaient été freinés par les priorités données face à l'arrivée de l'euro et à l'an 2000.

### ...EN BREF

## SGBD objet A vos ordres, SIR

● Versant, éditeur de bases objet, a été retenu par l'application SIR, le Système d'Information Régimentaire que Matra Systèmes & Information développe pour l'Armée de Terre. Ce système de commandement tactique comporte des règles de métiers, métiers d'armes. La cheffaine de projet, Clothilde Marchal, ne voulait pas que le projet comporte une gestion des tables relationnelles.

Tout est fait en objet, ce qui permet, en cas de bataille, d'avoir la réponse plus rapidement à une requête importante.

## Borland

● Borland est redevenu ce qu'il était, au sein d'Inprise. Une division portant ce nom mythique se charge à nouveau des outils de développement. Selon l'éditeur, le développement C++ classique est en régression de 20 % par an, tandis que le mode RAD reste stable, avec un marché deux fois plus important (cinq cent vingt millions de dollars). Ce sont les versions Pro des outils qui représentent la moitié du chiffre d'affaires, contre 15 % pour les versions standard et 35 % pour celles en client/serveur.

**...EN BREF**

**Médecins SOS SSII**

● Trop tard, les médecins sont déjà presque tous équipés, selon une enquête auprès de 600 médecins libéraux d'Ile-de-France, d'après l'union régionale des toubibs. 68 % d'entre eux ont déjà un PC et la majorité des non-équipés compte le faire rapidement. Le matériel acheté est récent et complet, avec même une part de marché juteuse pour Apple qui équipe 30 % des médecins. Il resterait un gâteau de 100 millions de francs pour les indépendants et SSII.

**Evaluation**

● COOL:Jex, environnement de développement et de modélisation orienté objet, a été choisi par Lockheed Martin, le célèbre fabricant de fers à repasser volants furtifs. Pour parvenir à un tel résultat, la firme américaine a mis en place un programme spécial pour créer une nouvelle race d'ingénieurs, aux chemises bien repassées, programmant en C++, Java, Corba IDL ou Ada, avec une parfaite maîtrise de la notation UML.

**Piratage**

● Selon BSA, le piratage coûte à l'industrie européenne quatre milliards d'euros par an. C'est bien connu, cela coûte cher d'être vénal.

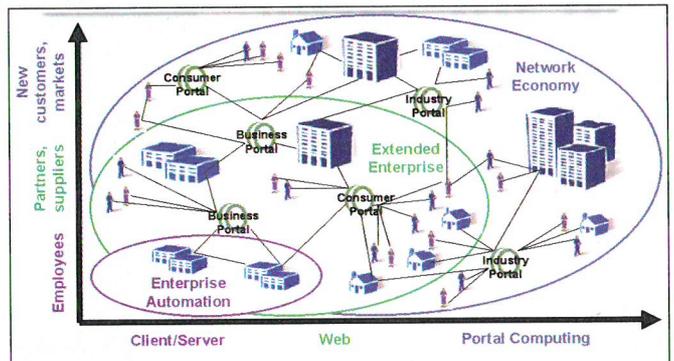
**Serveurs**

**Sun s'ouvre aux portails**

Après le rachat de NetDynamics, Sun annonce la version 5 de ce serveur d'application. Très musclé par son évolutivité et ses possibilités d'intégration, ce "fabricant de portail" est censé fédérer les clients, partenaires, fournisseurs et employés autour d'un nouveau modèle business client/serveur sur le Web.

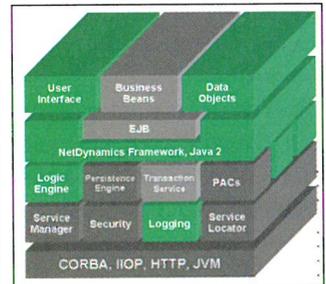
**Adapteur à brancher**

Dans la même position qu'IBM, en tant que fournisseur de hardware et de software serveur, Sun n'hésite pas à empiéter sur les plates-bandes de l'apôtre de l'e-business. Sur le plan technique, NetDynamics propose le PAC (Platform Adapter Component) pour brancher tous les différents systèmes entre

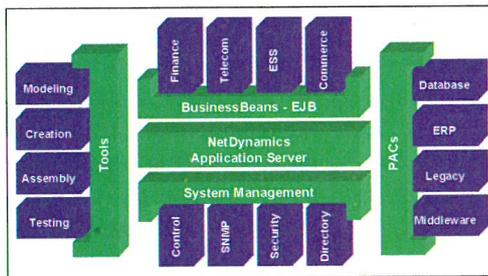


➤ Le portail NetDynamics fédère tous les business à la périphérie de l'entreprise.

eux. Le support Microsoft est intégral par les services de COM. Le support de CORBA et des JavaBeans est intégral, comme de bien entendu. Par ailleurs, le serveur supporte les nouveautés comme XML, mais aussi les références du terrain comme les protocoles mail, LDAP, CICS, IMS, MQSeries, AS/400... Plusieurs PAC seront



➤ Tous les standards web suivis.



➤ Une architecture modulaire pour brancher tous les systèmes, en recourant si nécessaire au SDK de l'éditeur, idéal pour les progiciels.

disponibles en option, notamment celui pour SAP et PeopleSoft. Un Hyper PAC SDK, permettra de raccorder des systèmes plus spécifiques. Il reste à savoir si tous les interlocuteurs quotidiens de l'entreprise voudront bien se faire PACsser le business. ■

**SYMPOSIUM**

**Java fait la fête**

La conférence Développeur Java se tenant les 27, 28, 29 avril au CNIT Paris-La-Défense est l'occasion de mieux connaître la dizaine de technologies-clés en langage Java et les solutions du marché. Le centre de conférence est presque entièrement réservé pour l'occasion. Les meilleurs spécialistes et praticiens viendront vous exposer leurs travaux pendant cette manifestation d'envergure



européenne. La principale priorité est de mieux valoriser les applications d'entreprise. ■ [www.jess99.com](http://www.jess99.com)

# File Maker pour le zéro papier

Le président de File Maker, filiale d'Apple s'appelant précédemment Claris, est un Français. Il est chargé de toute l'activité de développement officielle sur les Mac. Le produit phare est un solide SGBDR, agréable à visualiser. ■ par Alain Baritault

## ■ Qu'est-ce exactement que File Maker ?

C'est une "business unit" d'Apple créée par Steve Jobs il y a juste un an. C'est aussi un logiciel de gestion de base de données relationnelles. A l'exception de File Maker, un SGBD relationnel, la majeure partie du portefeuille de Claris repartait chez Apple, Mac OS, Claris Works... parce que ces logiciels étaient devenus stratégiques pour la compagnie. De fait, la nouvelle division renommée File Maker conservait un logiciel centré sur le secteur professionnel, représentant un portefeuille d'environ 70 millions de dollars, répartis pour moitié sur Mac et pour moitié sur PC. Ces derniers représentent cependant la croissance la plus forte en terme de chiffre d'affaires.

## ■ Quels sont ses atouts ?

Je pense que File Maker peut trouver sa place sur un marché dominé par quelques grandes compagnies. Techniquement, nous bénéficions de plus de recherche et développement totalement dédiés à File Maker, nos forces de vente sont plus spécialisées et se consacrent entièrement au produit. Aujourd'hui, notre chiffre d'affaires par trimestre atteint 20 millions de dollars, il y a 4 millions d'unités de File Maker installées dans le monde et nous sommes numéro 1 sur Mac, numéro 2 sur PC juste derrière Microsoft et devant

Corel. Sous Windows, nous avons doublé nos ventes en un an alors que Microsoft vend Access en bundle avec Office... Nous ciblons essentiellement les départements ou les petites entreprises qui n'ont pas accès aux ressources des gros services informatiques.

## ■ En quoi File Maker est-il bien adapté à cette cible ?

Je vois une opportunité pour File Maker dans les petits groupes de travail, parce que les services informatiques sont aujourd'hui largement encombrés par des problèmes liés au passage à l'an 2000 ou à l'euro. Ils se centrent par conséquent sur les activités critiques pour l'entreprise. Pour ce qui n'est pas critique, les utilisateurs ont deux solutions : prendre leur mal en patience ou résoudre eux-mêmes leurs problèmes. En général, ils n'ont pas besoin de grosses bases de données, ils savent ce qu'ils veulent mais ils ne

sont pas vraiment programmeurs. Ils veulent aussi l'interopérabilité avec les autres bases de données de la société. File Maker est le seul système de gestion de bases de données relationnelles pour non programmeurs. De plus, il est bon marché et se met facilement en réseau.

## ■ A quelles applications File

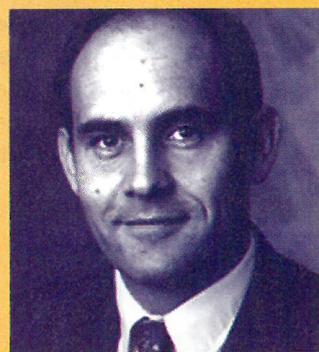
### Maker se prédestine-t-il, la bureautique ?

File Maker convient parfaitement pour toutes les applications de données personnelles. Mais ce segment est dominé par MS-Office. Nous ne cherchons donc pas à lutter contre Microsoft sur ce marché. Dans le secteur des entreprises, File Maker peut-être aisément utilisé comme Front End pour des grosses bases de données relationnelles comme Oracle ou Informix. Dans ce cas, File Maker peut supporter des applications

critiques pour l'entreprise. Les petites entreprises et les départements ont besoin de structurer et de partager des données individuelles, ils disposent de peu de ressources techniques et veulent que leurs applications soient développées très vite. Ils font souvent appel à des consultants experts dans un secteur vertical, qui utilisent File Maker pour sa facilité de mise en œuvre. File Maker est l'outil idéal pour remplacer les données papier, parce qu'il offre un mode de visualisation plus clair que dans les autres SGBD. C'est un "bon équipement de bureau".

## ■ Quelles sont les évolutions prévues dans File Maker ?

Nous avons noté que le commerce électronique est une activité en pleine croissance dans les entreprises. En complément du SGBD, nous avons aussi Home Page, un logiciel auteur pour le Web qui a acquis une bonne réputation de convivialité. Nous allons supporter XML et améliorer les fonctions de "Web Publishing" qui exigent une base de données et un dispositif middleware. En même temps, nous améliorerons les capacités d'intégration en respectant au maximum les systèmes informatiques existants des entreprises. ■



### ➤ Dominique Goupil

Président de File Maker

Il y a près d'un an, Claris, le principal éditeur de logiciels Macintosh, a changé de nom pour s'appeler Frame Maker, comme son fameux SGBDR. Une partie du catalogue initial a disparu pendant la métamorphose, mais Frame Maker rebondit en proposant une bureautique basée sur le SGBDR, avec une interface souple et conviviale.

“  
**Imposer File Maker sur un marché concurrentiel**  
”

Site officiel

■ [www.filemaker.com](http://www.filemaker.com)

# Technologies d'Etat

La MTIC est un centre de compétence, d'information et d'expérimentation, au service de l'Etat. C'est une sorte de SSII interne, visant l'amélioration des services et des coûts informatiques dans l'administration. ■ par Matthieu Thfoin

## ■ Depuis quand la MTIC existe-t-elle et quel est son cadre juridique ?

La Mission interministérielle de soutien technique pour le développement des Technologies de l'Information et de la Communication dans l'administration a été créée le 27 août 1998 par décret, pour une durée de trois ans. Sa mise en place s'est effectuée au cours de l'automne 1998. Elle est à présent opérationnelle, dans le cadre des missions qui lui sont confiées : maîtrise d'ouvrage, référencement technique, animation et diffusion d'informations technologiques. La mission pilote ainsi certains projets interministériels, comme l'Intranet de l'administration. Elle garantit la cohérence des référentiels techniques, accompagne le développement et favorise la mutualisation des savoir-faire. Enfin, elle impulse et initie de nouvelles pratiques. Toutes ces informations sont disponibles sur notre site Web. C'est dans l'ensemble de ce programme de travail que s'insère notre action sur les logiciels libres.

## ■ Pourquoi choisir Linux et non pas OpenBSD, FreeBSD ou NetBSD ?

Il ne s'agit pas à ce stade de choisir entre ces différents systèmes d'exploitation libres. La MTIC utilise ainsi pour son usage interne (Intranet) Apache sur FreeBSD, qui voisine fort bien avec

notre serveur Microsoft IIS sur Windows NT. La matinée *Linux et l'administration : premières pistes pour un usage raisonné*, que nous avons organisée le 15 janvier dernier, a d'ailleurs été l'occasion de rappeler la multiplicité des projets "libres" et la Foire aux Questions que nous préparons, avec le concours de nombreux internautes et de l'AFUL, portera sur Les systèmes d'exploitation libres et l'administration. L'accent a été jusqu'ici généralement mis sur Linux, mais je pense que l'ensemble des systèmes d'exploitation libres et plus largement des logiciels libres, profitent de cette notoriété. Au total, il me semble que le choix d'un logiciel, et notamment d'un système d'exploitation, doit relever d'une méthode industrielle.

■ **Le logiciel libre peut être une réponse technique et économique aux besoins des administrations, sans parler du côté politique. Laquelle de ces deux**

**problématiques est selon vous la plus importante : maîtriser à fond son informatique ou économiser au maximum pour un degré de performance correct ?**

Le mieux serait certainement de maîtriser complètement son informatique de manière

économique... Plus sérieusement, les besoins de l'administration en logiciels ne sont pas particulièrement différents de ceux des entreprises, à part dans certains secteurs spécifiques (recherche, domaine militaire...). Par ailleurs, les systèmes d'information de l'adminis-

tration ne sont pas monolithiques et il semble difficile de prétendre apporter une réponse unique à cette question. Il convient d'étudier, pour chaque processus, quel système est le plus à même d'apporter, pour le moindre coût, les fonctionnalités demandées, tout en préservant la cohérence globale du système d'information, ce qui ne signifie pas nécessairement un système d'exploitation unique.

■ **Comment les achats informatiques se décident pour les administrations ? Est-ce toujours le même processus de décision et s'avère-t-il suffisamment rapide pour être techniquement à jour ?**

Chaque ministère est responsable de sa politique informatique dans le respect des orientations interministérielles, notamment celles du Plan d'action gouvernemental pour la société de l'information. Dans ce cadre, il est notamment responsable de ses achats.

Il faut cependant relativiser le coût des achats de logiciels, qui ne totalisent qu'une partie relativement faible des budgets informatiques globaux dans toute organisation. Les décideurs publics suivent, en matière de passation de marchés informatiques, les mêmes règles que pour les autres marchés publics. Ces règles ont pour but de garantir une mise en concurrence ouverte et équitable des offreurs de solutions, sur les bases d'un cahier des charges fonctionnel. Elles se déclinent notamment en fonction des montants des marchés. Sur le fond, ces règles ne sont d'ailleurs pas autre chose que des principes de bonne gestion et elles sont similaires aux procédures d'achats des grandes entreprises. La durée du processus d'achat varie de quelques jours à quelques mois,



### ➤ Godefroy Beauvallet

Le Chargé de mission auprès du Premier ministre, dans le cadre de la MTIC, nous explique comment l'Etat veille aux meilleures technologies pour équiper les administrations et promouvoir leur connaissance. La pluralité des systèmes, choisis selon une méthodologie rigoureuse, place l'Etat devant les mêmes choix que toute entreprise. Rassurant !

ce qui là encore n'a rien d'atypique par rapport aux entreprises. Néanmoins, un certain nombre de situations concernant les marchés informatiques peuvent poser des problèmes, au demeurant partagés avec d'autres secteurs. Il s'agit notamment de l'achat des équipements à obsolescence rapide. Les logiciels libres n'échappent qu'en apparence à ces règles ; en effet, s'il n'y a pas achat de programme, il y a bien souvent achat de services, lesquels suivent les procédures normales.

■ La "libération" des systèmes d'information a des avantages, mais aussi des inconvénients. Ces derniers seraient en toute logique dûs au fait que tout le monde connaît les sources de linux, et le manque de localisation ou de prestations de services

Le fait que les sources de Linux soient publiques n'est pas nécessairement un inconvénient. En matière de sécurité, on a l'habitude de dire que "la transparence est un facteur de sécurité". Si des sources sont connues de tous, il y a une chance importante que ceux qui découvrent des failles de

sécurité soient des utilisateurs honnêtes, qui partageront cette information avec les autres. Bernard Lang, de l'INRIA, estimait ainsi le 15 janvier dernier que la publicité des sources est, au contraire, un avantage des systèmes libres. De façon habituelle, le manque de localisation et de prestations

de services est en revanche un problème des logiciels libres, qui les réserve à ce jour à des populations particulières ou à des usages spécifiques (ser-

veurs, stations de calcul...). Cependant, il semble que la situation soit en passe de s'améliorer avec la naissance d'un monde du service autour des logiciels libres. Il faut bien voir qu'il s'agit là d'une question tout à fait traditionnelle en informatique : qui a la base de développeurs et de prestataires de services et la plus importante ? Il importe donc de ne pas avoir une vision statique de ces questions. Par ailleurs, le cadre juridique fourni par les logiciels libres conduit à repenser les conditions d'appropriation de la compétence relative à ces logiciels.

■ **Finalement, pouvez-vous nous dire en théorie ce que coûterait une administration équipée 100 % commerciale et la même, 100 % libre ?**

Il m'est impossible de répondre à cette question. Mais faut-il lui accorder une

importance capitale ? Si je pense à un projet que je suis chargé de mettre en œuvre, qui concerne le partage d'informations entre les responsables de l'élaboration des textes réglementaires et législatifs, le coût d'achat des logiciels ne comptera que pour 10 à 20 % du coût total du projet. L'intérêt des logiciels libres réside donc moins dans la substitution terme à terme d'un logiciel commercial par un logiciel libre que dans leur efficacité pour diffuser des technologies innovantes, selon un modèle économique original et prometteur. C'est dans les effets globaux de l'irruption de ce modèle dans le secteur de l'informatique et, surtout, dans la mise en place de nouveaux rapports entre les utilisateurs et les offreurs de solutions, que peuvent résider des gains économiques importants. ■

**La mission**

- \* Présentation
- \* Evénements
- \* Programme de travail et projets
- \* La mission dans la presse
- \* Stages à la mission

**Vous recherchez des sites et des forums intéressants**

- \* L'administration et les autorités indépendantes
- \* Les associations professionnelles
- \* Autres liens

**Vous recherchez des**

- \* Référentiels techniques de l'administration
- \* Documents de référence de l'administration
- \* Référentiels, FAQs et RFCs
- \* Référentiels de l'IDA
- \* Lexiques et dictionnaires
- \* Outils à télécharger
- \* Informations sur le passage à l'an 2000

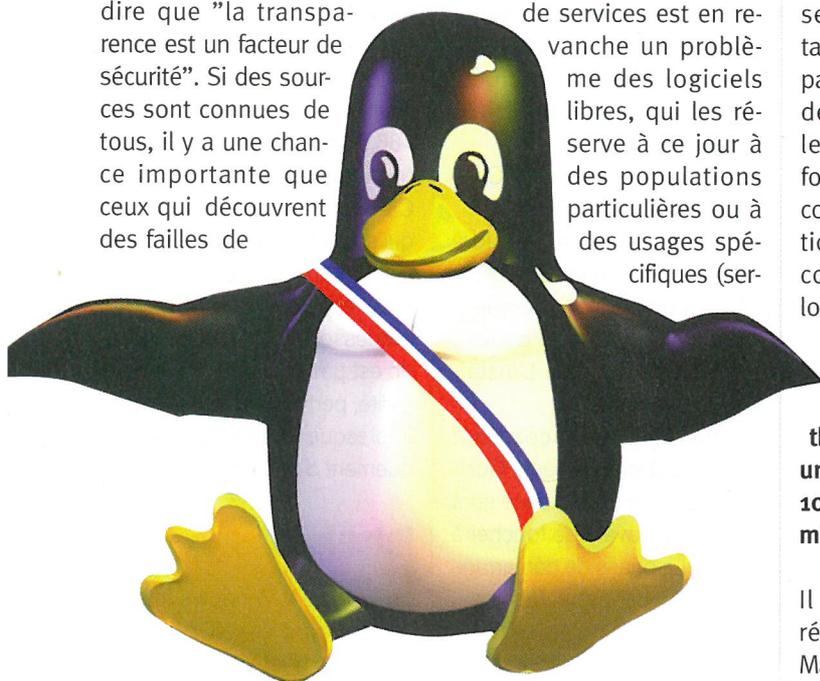
**L'actualité des T.I.C.**

**Plan du site et actualité**

**Contactez-nous**

Ecrivez au webmestre

**Les logiciels libres facilitent la diffusion rapide du savoir technologique.**





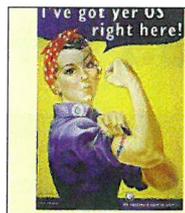
sur PC. Cette interface est d'ailleurs fluide et plutôt réussie, on n'est pas trop perdu. Cependant, cela ne vaut pas un AfterStep, une des meilleures si ce n'est la meilleure interface X-Windows sous Linux...

On notera immédiatement deux éléments originaux dérivés du menu Démarrer de Windows 95 : en haut, on trouve la liste des applications actives qu'elles soient en avant-plan ou en arrière-plan (background), ou encore dans les écrans virtuels. En bas, une grande barre sert un peu de trousse à outils : préférences, configurations... On y trouve un peu de tout.

## Novices à l'aise

Le KDE Control Center facilite grandement, pour les novices d'Unix, la configuration de l'OS. Les possibilités de personnalisation sont très nombreuses. LinuxPPC reconnaît plutôt bien les éléments du Mac : son, disque dur, CD-ROM, écran... Par rapport à MacOS, des plus incontestables : économiseur d'écran inclus (certains exemples donnent le frisson au vu des performances comme le KDE Science. Interface changeable à volonté grâce à la puissance incomparable des serveurs X-Windows, menus contextuels généralisés...

Revenons sur les écrans virtuels. LinuxPPC en offre quatre en standard comme le fait



[Home](#)

[About LinuxPPC Inc.](#)

[Products](#)

[Purchase LinuxPPC](#)

[Screenshots](#)

[LinuxPPC News](#)

[Press Releases](#)

**LinuxPPC** Linux for PowerPC™ Computers  
<http://www.linuxppc.org>

**Pre-order LinuxPPC 5.0!**

LinuxPPC Inc. develops and distributes the Linux operating system for PowerPC computers. Our flagship product, LinuxPPC 4.1, is the fastest OS available for PowerPC computers today, holding a 25-100% speed advantage over MkLinux and Rhapsody DR3.

**LinuxPPC Products: Release 5, Release 4.1, and Live 4.0.**

**LinuxPPC Release 5.0: The Next Big Thing**

LinuxPPC Release 5.0 will bring the latest Linux technologies - glibc 2.1, the 2.2 kernel, USB support - to PowerPC. R5 is still under development, and we do not have a release date set at this time.

> Le site Internet américain où le système LinuxPPC est disponible.

BeOS. Il s'agit d'espaces de travail indépendants les uns des autres. Sur l'écran A, on peut avoir un Netscape Communicator, sur l'écran B un traitement de texte, etc. C'est rapide et efficace. Il ne manque plus que le support multi-écrans comme son MacOS.

Les fenêtres sont en fait des fenêtres d'un navigateur Web comme sous Windows 98 et l'Active Desktop. On peut même configurer l'ensemble. La navigation peut se faire par icônes ou avec les boutons de la barre d'outils comme sous NextStep. Plusieurs modes d'affiches existent : explorateur de fichiers, icônes... Tous ces modes sont combinables.

On l'avait dit dès le départ, LinuxPPC live est un OS puissant, les benchmarks processeurs sont là pour le certifier. De nombreux utilisateurs Mac rêvent des fonctions modernes que l'on retrouve sur MacOS X

Server : multitasking, micro-noyau, multitâche préemptif, multithreads, gestion de la mémoire dynamique, multiples lancements de la même application, toolbox ré-entrante, serveur X-Windows, 100 % natif PowerPC ou encore multi-utilisateur. LinuxPPC, même en live, impressionne par ces fonctionnalités. Surtout lorsque l'on pense que son coup est dérisoire, voire totalement gratuit en téléchargement.

LinuxPPC exploite totalement le PowerPC, les logiciels 3D n'en seront que ravis. Le système est d'une réactivité dépassant et de loin, un Windows NT.

## LinuxPPC live en pratique

Un seul mot à retenir : wouah !!! On maîtrise rapidement les bases de LinuxPPC, l'interface KDE y aide beaucoup. Mais malheureusement, ainsi que le précise l'éditeur, LinuxPPC n'est qu'une démonstration des possibilités de LinuxPPC. Cette simple démonstration nous laisse tout de même un sentiment amer. Car pourquoi LinuxPPC Inc. n'a-t-il pas proposé un live 100 % opérationnel en

supprimant tous les codes sources et les accessoires ? C'est bien le problème de la version live : on reste sur notre fait. Pas de lecteur de disquette, impossible de reconnaître les CD-ROM, absence des aides, fonctionnalités réseaux incomplètes... Dommage ! L'implémentation du lecteur de disquettes aurait été un bon moyen pour travailler.

D'autre part, on s'aperçoit que la configuration que l'on fabrique n'est pas sauvegardée... Il est vrai que la version ne fait que 100 Mo et qu'elle fonctionne directement sur une partition HFS, mais on aurait attendu quelques fonctions de base supplémentaires.

Finalement, LinuxPPC 4 live n'est qu'une attraction technologique pour attirer les utilisateurs vers les versions complètes, celle de 600 Mo.

Malgré tout, la solution live n'est-elle pas adaptée aux novices qui ne veulent pas partitionner leur disque dur et avoir un minimum de chose à faire ? ■



**Site officiel**

■ [www.linuxppc.org](http://www.linuxppc.org)

### POUR

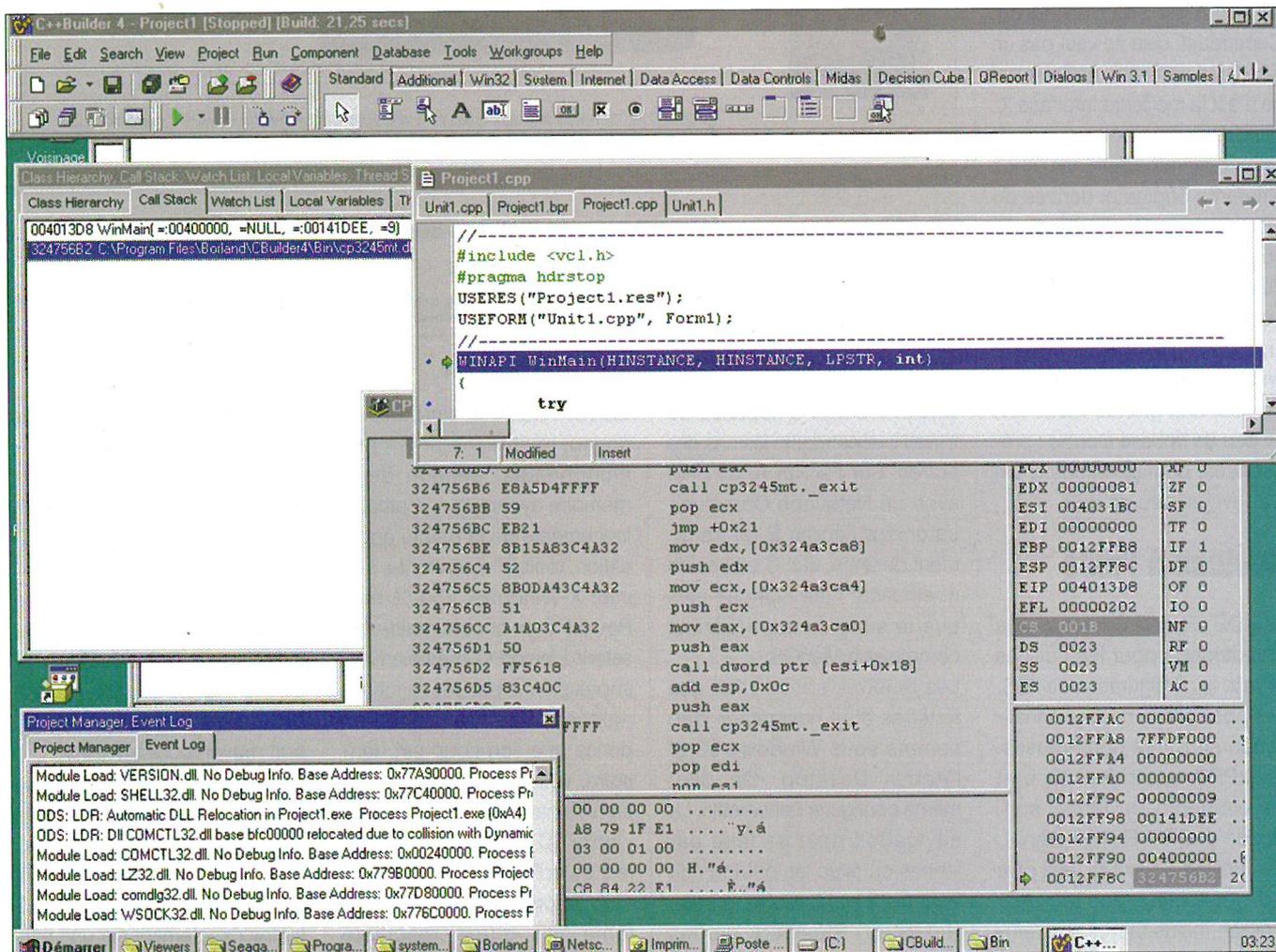
- Fonctionnement live.
- Natif 100 % PowerPC.
- Fonctions modernes.
- Interface KDE.
- Puissance brute.

### CONTRE

- Manque de fonctionnalités de base.
- Pas d'aide.
- Quelques instabilités.
- "Aspect démo".

INPRISE C++ BUILDER 4

## Le RAD idéal pour tous les C++



> Le merveilleux débogueur en action affiche toutes les informations-clés comme les variables locales, threads, piles...

**Quel développeur C++ n'a jamais entendu parler de C++ Builder d'Inprise ? Acteur incontournable, cet environnement allie allègrement le concept de RAD à la puissance du langage C++.**

La toute nouvelle version 4 apporte un grand nombre de nouveautés. Une des plus importantes est la compatibilité avec l'an 2000 (variable `TwoDigitYearCenturyWindow`). C++ Builder se veut désormais plus ouvert. Si les Active X étaient disponibles dans les versions plus anciennes, ils disposent maintenant des liens OLE et OLE Automation. Les composants COM sont eux aussi acceptés. Toujours dans la catégorie composants distribués, C++ Builder inclut

CORBA, que ce soit en serveur ou en client. Inprise veut suivre de très près le marché des SGBD. Pour ceux qui développent avec Visual C++ de Microsoft, C++ Builder facilitera le transfert des projets. Un convertisseur de projets est à votre disposition (utilitaire `VCToBPR`)...

C++ Builder possède une panoplie de projets dépassant tous les rêves ou presque... Eh oui, s'il s'agit avant tout d'un environnement de développement, il saura répondre à tous les

besoins d'un développeur ou d'un éditeur : application stand alone, batch, DLL, application serveur pour le Web, librairie MFC, Active X, Corba... on peut créer tout ce que l'on souhaite... Ainsi, tous les développements peuvent être centralisés dans un seul et même environnement. Avantage non négligeable si une application stand alone doit inclure DLL, composants ou connections SGBD. Par contre, si C++ Builder s'ouvre à de nouveaux horizons, il se refuse encore à Java même

dans l'intégration d'applets. Il est vrai que l'éditeur possède son propre RAD Java avec Java Builder. Malgré tout, on pourra utiliser l'ensemble Tool.h++ Professional permettant une communication avec Java. Mais cela ne va pas plus loin.

## Look and feel

Vous avez déjà, sans doute, développé en Visual Basic ou en Delphi ? C++ Builder ne vous surprendra donc pas !!! L'ergonomie est quasiment la même. Barres d'outils en haut, fenêtre des propriétés à gauche... Les objets sont très nombreux et couvrent tous les domaines : objets d'interfaces, base de données (SQL, états, requêtes),

Internet (sockets, URL, messagerie)... De quoi bâtir des interfaces solides très rapidement. Les propriétés ne fournissent qu'un minimum de possibilités. Une partie du code est automatiquement généré, dans les événements par exemple, mais cela reste très restreint...

Concernant le Project Manager, même si le Glisser-déposer est autorisé, il n'offre que peu de fonctionnalités... Pour les autres fenêtres de C++ Builder, on retiendra bien entendu le navigateur de classes. C'est un bon moyen pour retrouver la hiérarchie entre les différentes formes qui ont été créées. Si le navigateur ne plaît pas, on se rabattra sur la fenêtre Class Hierarchic. Sur les options de

configuration, elles sont nombreuses aussi bien sur l'environnement de C++ Builder que sur le débogueur.

En parlant de débogueur, il s'agit d'une des références actuelles dans le monde PC. Ultra complet, le développeur a à sa disposition toute une série de fenêtres debug, chacune remplissant une fonction précise : variable locale, threads, piles... On passe d'une à l'autre pour observer le moindre détail. Si besoin était, la fenêtre CPU décortique en adresses assembleur le code exécuté.

Raffinement suprême ? La fenêtre de Log inscrivant à la volée toutes les actions réalisées. Ce qui permet de repérer des bugs particulièrement

"vicieux" dans leurs déclenchements. Après une telle liste, est-il utile de préciser qu'il y a le mode trace, en pas à pas avec points d'arrêts...

Difficile de critiquer un tel environnement de développement. On peut tout de même émettre une réserve sur une astuce d'interface dockable, qui consiste à "encapsuler" des fenêtres dans une "fenêtre mère" pour gagner de la place et, surtout, pour tout rassembler en un seul endroit. Si tout cela est pratique pour les options du débogueur, pour les autres, le choix n'est pas forcément très justifié...

Une question de goût. ■

## Site officiel

■ [www.inprise.fr](http://www.inprise.fr)

# Rejoignez les utilisateurs avertis



Chaque mois, bénéficiez de l'essentiel de l'information pour être à la pointe de la technologie

- Les actualités exclusives
- Les fiches et les conseils pratiques
- Les tests et les bancs d'essai



**Sur le CD-Rom :**  
toute l'actualité  
logicielle pour  
votre PC !

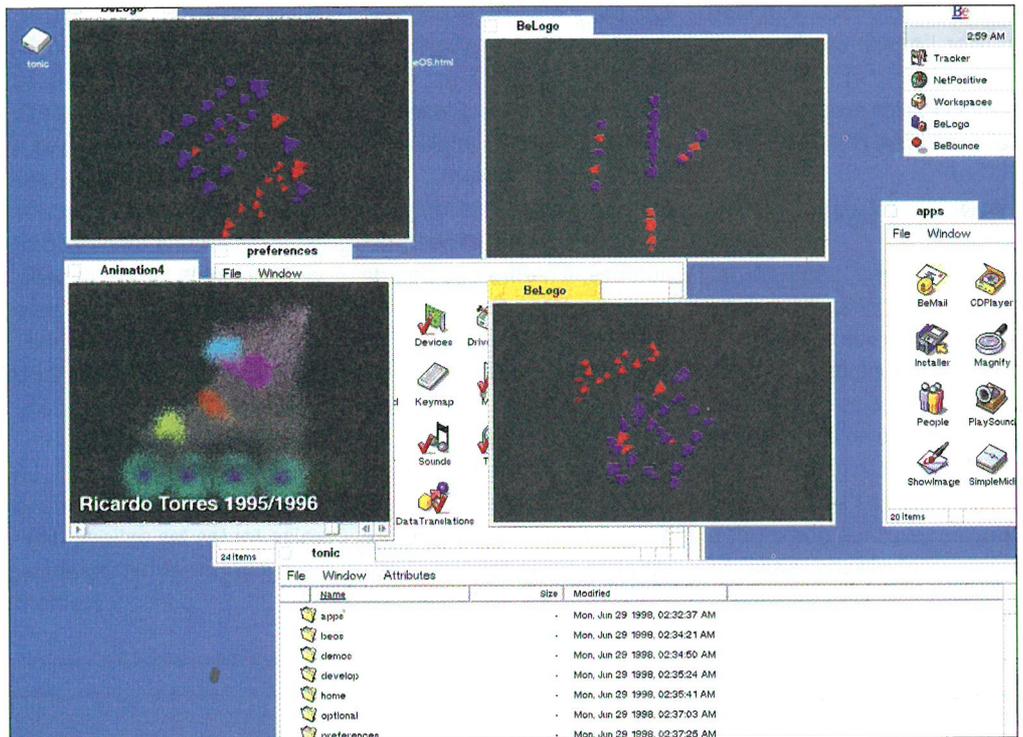
EN VENTE LE 25 MARS CHEZ VOTRE MARCHAND DE JOURNAUX

## BEOS R4

# L'OS qui avait la beauté en plus...

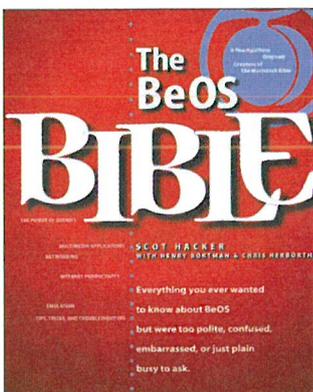
BeOS peine à casser son image de gadget ultra-technologique.

Pour prouver ses qualités, Be a sorti une version 4 confirmant l'orientation Intel du système. Y-a-t-il encore une place pour lui sur le marché cruel des PC ? On lui décerne volontiers la palme des services développeurs



> Vue d'ensemble du bureau de BeOS r4, avec quatre fenêtres de vidéos indépendantes et fluides.

Après une version 3 pour Intel un peu faible et un manque d'intégration aux PC, la release 4 comble un grand nombre de lacunes. Si le nombre de périphéries reconnues est en nette hausse, mieux vaudra s'armer d'un PC récent conforme Intel pour éviter des problèmes. Un bon Pentium



> Un des ouvrages anglais recommandé par l'éditeur de BeOS.

350 avec 64 Mo de ram suffira amplement à exploiter la bête. L'installation se déroule toujours en deux étapes : partition avec Partition Magic puis installation du système en lui-même. Un peu rebutant et peu rapide... Bien que l'on puisse installer à partir d'un Windows 95, 98 ou NT, NT pose plus de problèmes... Dommage de n'avoir pas unifié une fois pour toute l'installation sans avoir à redémarrer plusieurs fois... Petit point étonnant, si on laisse le CD-ROM système, au redémarrage, il recharge l'application d'installation. Allez, encore un petit effort pour la r5 ?

Passons aux choses sérieuses. Le démarrage système est plutôt rapide. On retrouve un bureau identique à du MacOS. Le seul élément original est le

de contrôle pour voir toutes les applications actives. Accéder au A propos, l'outil de recherche ou encore pour quitter BeOS... Les menus contextuels sont partout avec de très nombreuses options.

### Y a-t-il un pilote...

Le fait que le système soit multitâche préemptif offre des performances multimédias hors du commun sur PC, que ce soit en 3D ou en films QuickTime !!! Cinq ou dix animations, peu ou prou de ralentissements... Décidément, le multimédia semble être le point fort de BeOS. Les démonstrations livrées terminent de convaincre les amateurs.

Comment se débrouille BeOS dans le monde PC ? Côté performances pures et fonctionna-

lités, on retrouve exactement ce que l'on trouve sur PowerPC. De ce côté-là, aucune surprise à avoir donc. Si les chaînes IDE et SCSI sont mieux reconnues, le manque de pilotes se fait sentir. Mieux vaut posséder les standards du marché ou lire les listes de compatibilités... Bien que Mount voit les partitions NTFS, BeOS a du mal à les monter. Côté réseau, il faudra faire preuve de patience pour pouvoir monter son réseau. Les développeurs et autres amateurs de nouvelles technologies ne manqueront pas d'apprécier. Débogueur, sources, exemples par dizaines avec en prime un BeIDE, un Code-

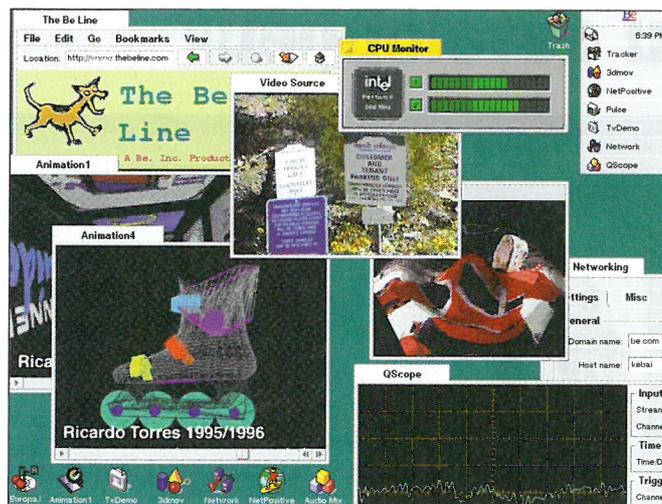
### Site officiel

- [www.be.com](http://www.be.com)
- [www.bedepot.com](http://www.bedepot.com)

Warrior pour BeOS concocté par Metrowerks. Les développeurs sous CodeWarrior retrouveront leurs habitudes avec une interface très proche de son grand frère. L'environnement est une chose, mais le plus attrayant dans BeOS, c'est un système entièrement objet !!! Une seule chose manque à l'appel, Java. Pour le moment, Java n'est pas implémenté au cœur de BeOS. Etant disponible sur Intel et PowerPC, il suffira de compiler sur chaque plateforme le code de son application. Toutefois, il ne faudra pas oublier de consulter les notes techniques pour respecter les structures spécifiques aux plateformes.

## Ravalement de façade

L'interface en elle-même ne ressemble à rien que l'on connaisse. Le design est original. Par contre, l'ergonomie manque encore à l'appel. Les options des menus contextuels du Bureau risquent de surprendre. Pourquoi ? Le menu Desktop



> Les capacités du multimédia sous BeOS demeurent une référence pour la qualité des API disponibles.

sert aussi de mini-navigation dans les volumes... Car on oublie que BeOS n'a pas de barre de menus standard comme Windows ou MacOS. Cette absence se sent dans les fonctions de base du Bureau... Si on peut monter automatiquement les volumes sur le Bureau, les rafraîchissements laissent par contre à désirer. Pour les disquettes, il faut remonter le volume si on les déverrouille...

Un autre exemple ? Si on veut changer le mode d'affichage, de liste à une vue par icône, la fenêtre reste vide... On est alors obligé de revenir au mode liste. Incontestablement, BeOS

r4 fait oublier la précédente version. Le système est bien plus stable et plus performant. Le multitâche impressionne par sa capacité à supporter une multitude d'animations simultanément...

Seul petit manque dans les fonctions avancées, le multi-utilisateur. L'orientation objet de BeOS, l'environnement BeIDE et les exemples fournissent aux développeurs une aide précieuse. L'ergonomie, le manque de logiciels et de pilotes sont les principaux reproches que l'on puisse faire, sans oublier quelques bugs ici et là, plus ou moins voyants.

L'installation, bien qu'améliorée, manque toujours de simplicité surtout sous Windows NT. Malgré tout, les capacités multimédias valent largement le détour. ■

### POUR

- Système objet.
- Fonctions modernes.
- Meilleur support PC.
- Multimédia.

### CONTRE

- Installation sous Windows NT.
- Ergonomie.
- Bugs.



# SNiFF+™

## L'ABC du code source !

Analyse, Browsing, Compréhension

**SNiFF+, un seul outil pour vous orienter dans votre code source :**

- 1<sup>er</sup> environnement SCE (Source Code Engineering)
- Multiplate-forme (UNIX, Windows)
- Multilangage (C/C++, Java™, Corba IDL, Fortran, Ada, ...)

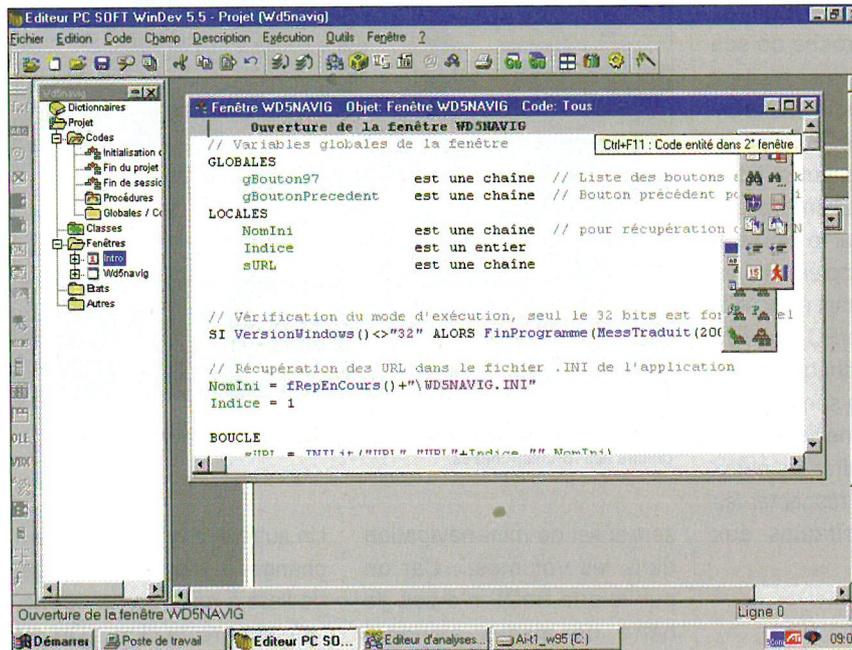
[www.takefive.com](http://www.takefive.com)

**TakeFive**  
software  
An Integrated Systems Company

## WINDEV 5.5

# L'intégré du développeur

Héritier du mythique High Screen sous DOS, WinDev s'inscrit dans le "tout en un" du développeur. Il essaye de rationaliser en un unique environnement tous les modules de développement.



> Éditeur de code de WinDev 5.5 au premier plan. La fenêtre du gestionnaire à gauche.

**W**inDev intègre une impressionnante quantité de modules et de fonctions. Le dongle, clé de sécurité, est malheureusement toujours de rigueur pour le faire fonctionner. On peut distinguer trois grandes parties dans WinDev : l'analyse (avec Merise), le RAD et la base de données. Bien que WinDev sache travailler avec du code extérieur et d'autres bases de données, l'environnement reste propriétaire par son langage et sa base de données, Hyper File. Du côté du Web et de Java, les choses se gâtent un peu. Si WinDev est capable d'inclure

des applets Java, il est hors de question de développer le moindre code Java ou la plus petite interface Swing. En tout cas pour l'instant... De plus, les applications générées par WinDev ne sont pas telles quelles transférables sur le Web. Pour cela, il faudra acquérir le module WdWeb. Cet utilitaire de WinDev transforme votre projet WinDev pour Web avec pages HTML et code CGI. Quelques nouveautés remarquées de la version 5.5 : les fonctions Internet intégrées (messagerie, manipulations sur serveur FTP), an 2000, Euro. Pour les utilisateurs d'Active X,

un navigateur ActiveX est disponible. Autre nouvel utilitaire, le WdInd, pour générer des réindexations à partir d'un fichier Hyper File : cependant, son champ d'action est limité, il travaille à partir d'une indexation existante.

### Un certain manque de convivialité

Côté codage. Une partie du code est automatiquement générée. La syntaxe est un mélange de Basic, de Pascal et de C, en français, ce qui risque d'en troubler plus d'un. On ne peut pas dire que l'ergonomie des fenêtres de code soit particulièrement réussie, c'est très sobre, trop sans doute. C'est une impression générale d'ailleurs. L'interface et l'ergonomie de WinDev manquent de personnalité et de qualités. Okay, c'est une interface Windows classique mais elle est plus proche du Windows 3.1...

Un brin de convivialité aurait été le bienvenu. On peut dire la même chose pour le débogueur. Il offre les fonctions de base, mais sans plus. Par contre, WinDev garde la compatibilité avec Windows 3.1 par la compilation 16 bits, fonction capitale pour certaines sociétés. Il sait aussi travailler en groupe, un fonctionnement réseau est possible avec sécurité d'accès par mot de passe. Si WinDev représente une bonne alternative aux environnements habituels, le côté propriétaire, le manque de convivialité et d'intégration des modules, sans oublier, la présence minimale de Java, sont autant de handicaps pour cet environnement. Malgré tout, WinDev sera parfait pour des entreprises ou des SSII qui souhaitent développer rapidement des projets sur mesure. ■

#### POUR

- Les différents modules.
- La documentation.
- Compilation 16 bits.
- Active X.
- Développement réseaux.

#### CONTRE

- Interface médiocre.
- Support Java.
- Clé électronique.
- Trop propriétaire.

#### Site officiel

www.pcsoft.fr

# L'exploitation des tables Paradox

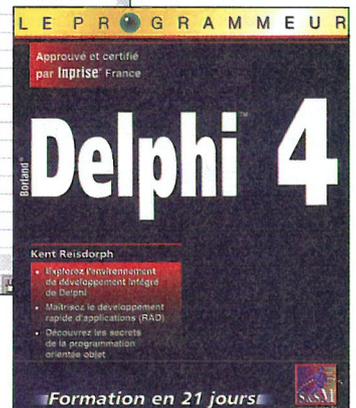
Les tables Paradox nous offrent un certain nombre de particularités qui doivent être connues pour optimiser leur exploitation.

■ Par Jean-Paul Pruniaux



ID	Clef	Titre	Nom	Prénoms	Adresse	Adresse 2
1	AOSH		Boyd	Eva		
2	AOSH MAA	Mme	(AOSH MAA)	Catherine ABEL		
3	AOSHCALL	Mme	(AOSH)	Sylvie		
4	ARMELLE	Me	MARREC	Armelle	43A, Quai Robert ALBA	
5	BLAISE	Mme	Blaise	Monique	6, rue des Frères Maire	
6	BOURLIL	Dr	Boubill	Claude	117 bis rue Ordener	
7	CHARLES	Mme	Charles	Yves & Anne-Henriette	7, Chadbourn Road	Lexington - Massachuset
8	COOKER	Mme	Cooker	Patricia	17, rue Collette	
9	DARE	Mme	Daré	Marion & Bernard	17, rue des Romains	
10	DELAFRAY	M	Delafraje	Her	107, av du Général de Gaulle	
11	DENIS	Mme	Denis	Fies	rue Legendie	
12	DENTISTE	Dr	Barnet	Ser	Passage Courtois	
13	ECOLEEVE		Ailette Sanguinetti	Did		
14	ESCUDIER	M	Escudier	Did		
15	HELENE	Mme	Péron	Héli		
16	JCM	Mme	Malhey / Scouamec	Jea	av de la Porte Chaumont	
17	KAMI	M	Capron	Michel	14, rue Claude de Bussy	
18	MALOEUVF	M	Maloeuvre	Stéphane Maloeuvre		
19	MARTY	M	MARTY	Samuel	56, rue des Sablons	
20	MATARAZZ	M	Matarazzo	Denis		
21	MCHAREK	M	M'Charek	Chokri		
22	MIALARET	M	Mialaret	Charles	28-30 rue Marcel Dassault	
23	MONNIE	Dr	Monier	Françoise	111, Bd Beaumarchais	
24	NORDLANE		(Nordland)			
25	PETIKA	Mme	Pélika	Tony et Corine		
26	PRUNIAUX	Mme	Pruniaux	Georges et Annie	1, La Gallardeie	

> La base de données Paradox a encore des émules et s'interface avec Delphi.



> L'un des ouvrages de référence par Campus Press.

Parmi les caractéristiques des tables Paradox, nous rencontrerons certaines précautions à prendre dans un environnement réseau, ainsi que des capacités plus méconnues comme celles des index décroissants. Tout ceci se résout en dialoguant avec le BDE par les API adéquates.

## Mettre les erreurs à l'index

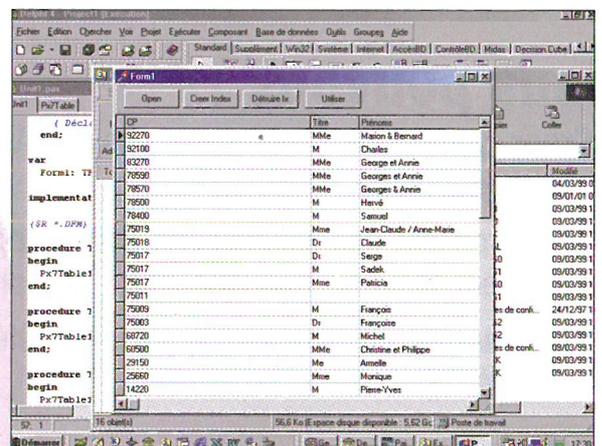
Dans nos articles précédents, nous avons fait à plusieurs reprises référence à un problème qui nous est arrivé quant à l'utilisation de tables Paradox en réseau, avec une erreur survenant parfois : "Index

out of date" (un index n'est pas à jour). Manifestement pour une raison x ou y, notamment après le plantage ou la fermeture abusive d'une station, certains index ne reflètent plus l'état de la table. Pour remettre en route l'application, il faut d'abord reconstruire les index avec l'utilitaire adéquat (voir entre autres notre précédent article). Nous avons alors suivi les directives proposées par un de nos correspondants sur CompuServe et l'incident a pour ainsi dire disparu. Voici la traduction de ces directives, proposées par "Reinhard Kalinke", (*r\_kalinke@compuserve.com*) : La plupart des problèmes 'index out of date' sont issus de terminaisons anormales de programme (coupure de courant, utilisateurs éteignant leur station sans fermer correctement leur

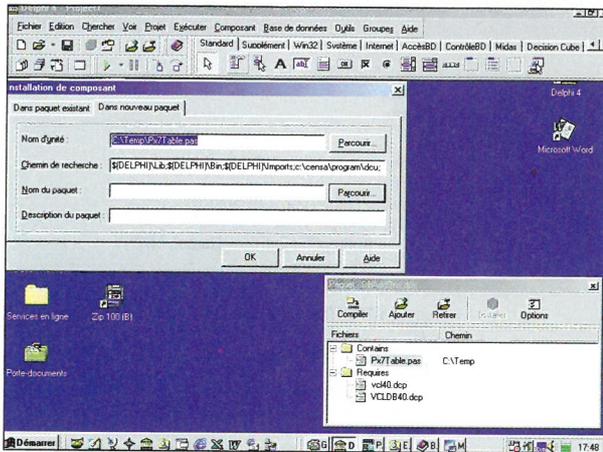
application, etc.). Les erreurs peuvent également être provoquées par de mauvaises configurations ou par du matériel réseau défaillant (câbles, cartes etc.).

## Sites de l'auteur

- JP\_Pruniaux@compuserve.com
- jpruniau@planetec.fr
- [http://ourworld.compuserve.com/homepages/JP\\_Pruniaux](http://ourworld.compuserve.com/homepages/JP_Pruniaux)



> Un formulaire Delphi, doté de fonctions sur les tables et index de Paradox.



> L'installation du composant qui gère Paradox dans l'environnement de développement de Delphi.

Voici ce que vous pouvez faire dans votre application :

- Créer un gestionnaire d'événement "OnAfterPost" pour chaque table de votre application et émettre une commande "DBISaveChanges(TableX.Handle)", où "TableX" représente la table concernée. Ceci demande au BDE d'écrire tous ses buffers concernant cette table. Malheureusement, les tables DBF (dBase) ne répondent pas à cette commande comme le font les tables DB (Paradox). Avec les fichiers DBF, les buffers ne sont enregistrés que lorsqu'on les ferme.

- Désactiver au niveau du système d'exploitation toute écriture en amont du cache (write-behind caching) de manière à ce que "DBISaveChanges" écrive vraiment sur le disque au lieu du cache disque.

- Intégrez à votre application une gestion simple de "login" et de "logout" (en gardant la trace dans un journal) de manière à identifier tout utilisateur qui ne ferme pas correctement son application. Ces informations pourront être utilisées pour éduquer l'utilisateur et pour montrer à votre client comment les problèmes peuvent arriver.

- Fermez explicitement vos tables dans l'événement "On-CloseQuery" de votre fenêtre principale. Vous pouvez remarquer que l'événement "OnClose" de la fiche principale n'est pas appelé lors de la fermeture d'une fenêtre ; l'événement "OnCloseQuery" d'une fiche secondaire n'est appelé que lors de la fermeture de la fenêtre principale ; l'événement "OnCloseQuery" de la fiche principale

est toujours appelé.

- Assurez-vous que le programme exécutable et les données sont stockés dans des répertoires différents.

- Voici d'autres remèdes spécifiques au système :

**NT** : désactivez "Opportunistic Locking" dans les registres.

**WGFW** : désactivez l'accès 32 bits au fichiers

**Win95** : désactivez "new file locking scheme."

(Performance/Dépannage)

Si les erreurs persistent,

vous devriez essayer et vérifier la configuration du réseau et le matériel. Les fichiers de bases de données supportent très mal les erreurs de transmission (...), si un octet d'un fichier de base de données est mal transmis, vous pourriez ne plus être capable d'accéder à cette base de données. (...)

Tout ceci représente des informations de référence qui nous ont permis de résoudre la plupart des problèmes que nous avons rencontrés. Nos recherches sur le sujet nous ont permis de détecter également une autre information-clé. Dans le paramétrage du BDE, il faut s'assurer que le flag "LOCAL SHARE" vaille "TRUE" pour la gestion des tables Paradox (se reporter à notre précédent article sur la création d'utilitaires de configuration du BDE).

## Les index décroissants

Les utilisateurs de Paradox en tant que tel connaissent ces index décroissants, mais qu'en est-il des programmeurs Delphi ? Et bien, les "TTable" standard ne permettent pas d'y avoir accès et seul "Reinhard Kalinke", là encore notre sauveur, offre des solutions. Il propose notamment un fichier "Px7Table.pas" à télécharger. Ce fichier "Px7Table.pas" définit un nouveau composant, "TPx7Table", qui hérite de la classe "TTable".

Ce composant, qui peut se comporter comme un "TTable" standard, possède des fonctionnalités complémentaires qui permettent, entre autres, la création d'index secondaires décroissants. Une fois

que ces index ont été créés avec le "TPx7Table", la table peut être ouverte par l'intermédiaire de n'importe quelle "TTable" qui saura reconnaître ces index décroissants.

Vous trouverez page 26 un extrait du listing de ce composant proposé par "Reinhard Kalinke". Nous avons épuré de ce dernier tout ce qui ne concerne pas la gestion d'index secondaires. Pour comprendre le fonctionnement de ce composant, consultez l'aide des API du BDE pour tous les termes que nous avons mis en gras dans ce listing.

Ci-dessous, vous trouverez le source d'une petite application qui teste le composant "TPx7Table" une fois qu'il a été installé (menu "Composant / Installez un composant"). ■

## Application du test

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Classes,
  Graphics, Controls, Forms, Dialogs,
  StdCtrls, Db, DBTables, Px7Table, Grids,
  DBGrids, ExtCtrls;
type
  TForm1 = class(TForm)
    Panel1: TPanel;
    DBGrid1: TDBGrid;
    Px7Table1: TPx7Table;
    DataSource1: TDataSource;
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
  private
    { Déclarations privées }
  public
    { Déclarations publiques }
  end;
var
  Form1: TForm1;
implementation
{$R *.DFM}
procedure TForm1.Button1Click(Sender: TObject);
begin
  Px7Table1.Active := not Px7Table1.Active;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
  Px7Table1.AddPx7Index('Test', 'CP;Nom', 'CP',
    [ixDescending]);
end;
procedure TForm1.Button3Click(Sender: TObject);
begin
  Px7Table1.DeleteIndex('Test');
end;
procedure TForm1.Button4Click(Sender: TObject);
begin
  Px7Table1.IndexName := 'Test';
end;
end.
```



# Organisez l'info avec un outil de pro

Vous avez tous entendu parler de WorkFlow. Que signifie ce mot après tant d'années ? La présentation d'un exemple simplifié, à travers l'outil de **PeopleSoft**, vous éclairera tout en vous soufflant quelques idées pour votre entreprise.

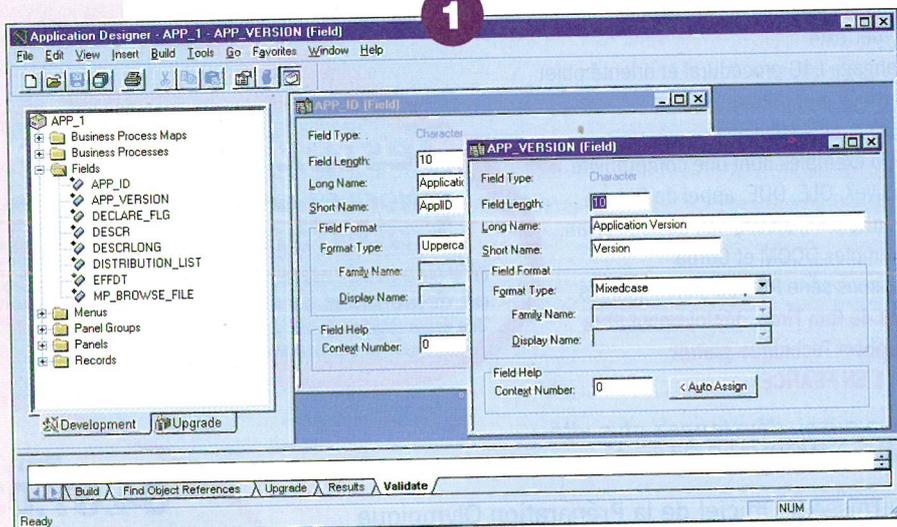
■ Par Marc Perroulaz  
mperroul@ctm.ch



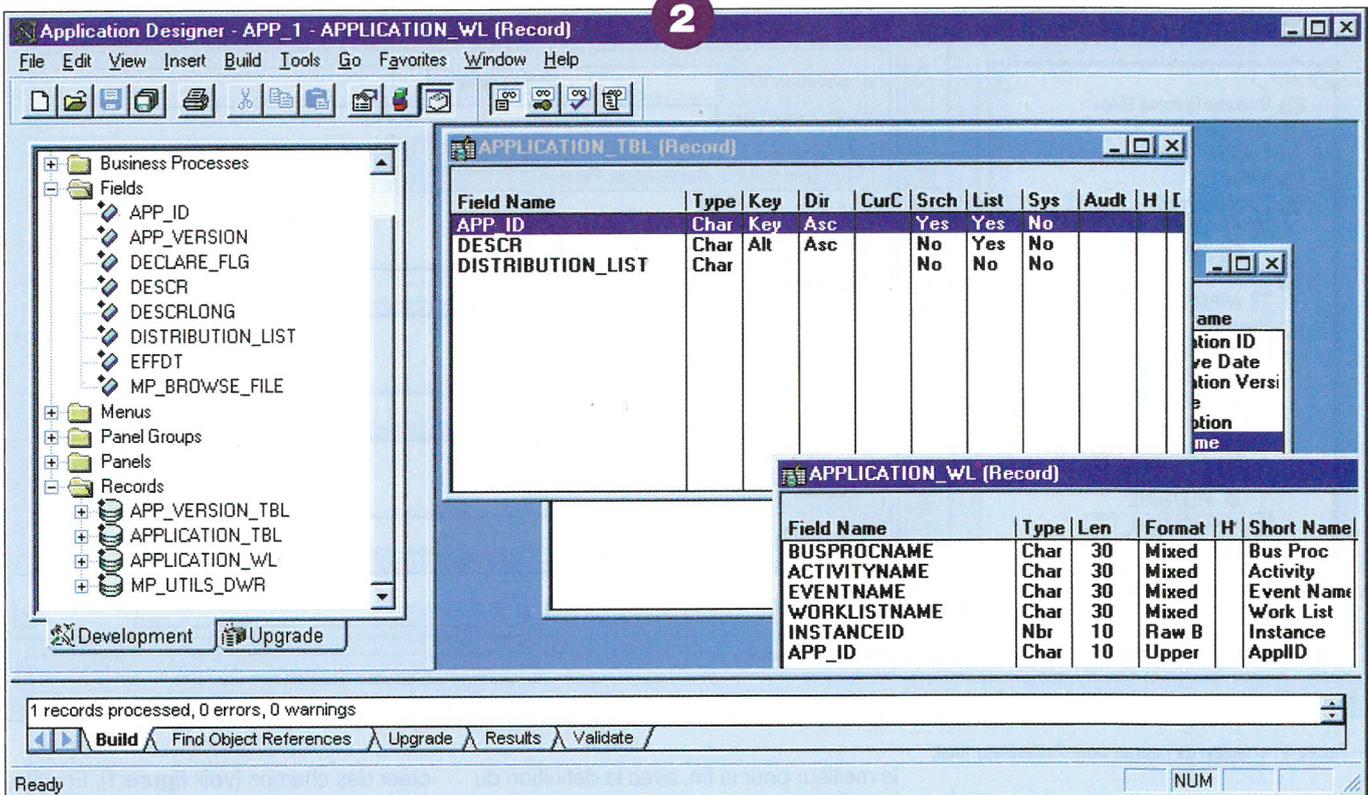
Né il y a bien longtemps, le WorkFlow peut se traduire par :

- une suite de procédures automatisées,
- un enchaînement de modules pré-programmés,
- un partage d'informations sécurisées,
- un contrôle de flux d'informations,
- une automatisation des liens inter-tâches.

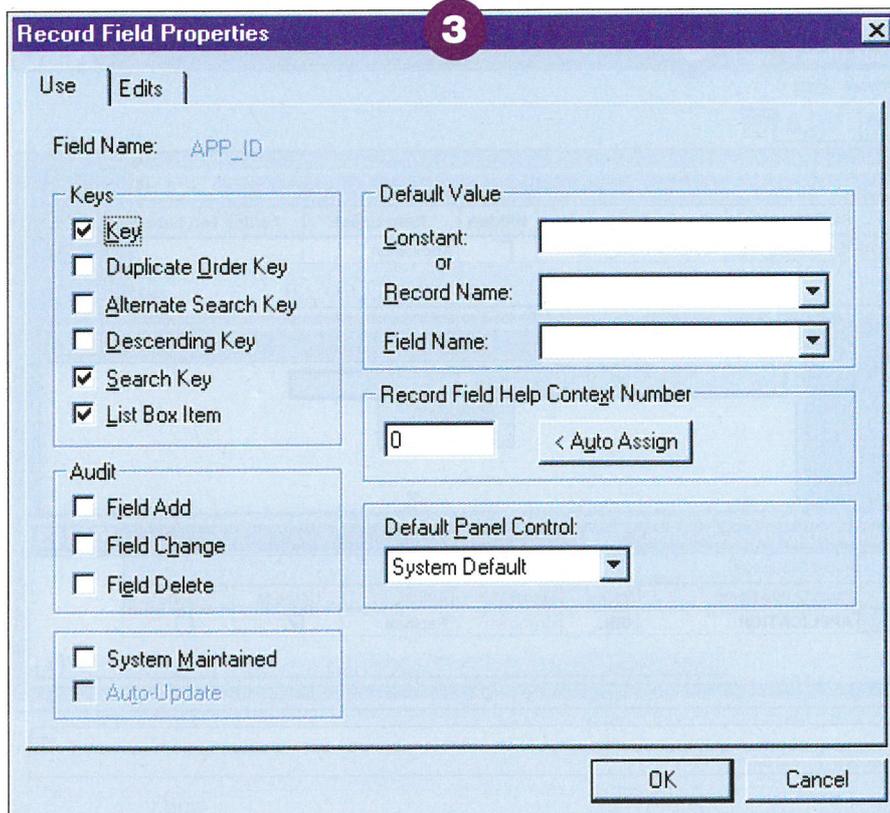
Nous n'allons pas débattre de chaque définition. N'importe quel programme contient un ou du WorkFlow, souvent transparent pour l'utilisateur. Il est clair qu'un logiciel peut être basé sur un moteur de WorkFlow, surtout pour de la GED. Le moteur se chargera alors de déplacer l'information d'un bout à l'autre d'un processus défini avec l'utilisateur.



> Figure 1 : création des champs dans le concepteur d'application de WorkFlow.



> Figure 2 : création des tables de données dans le concepteur d'application.



> Figure 3 : définition détaillée d'un champ, ici aux propriétés recherchables, listables...

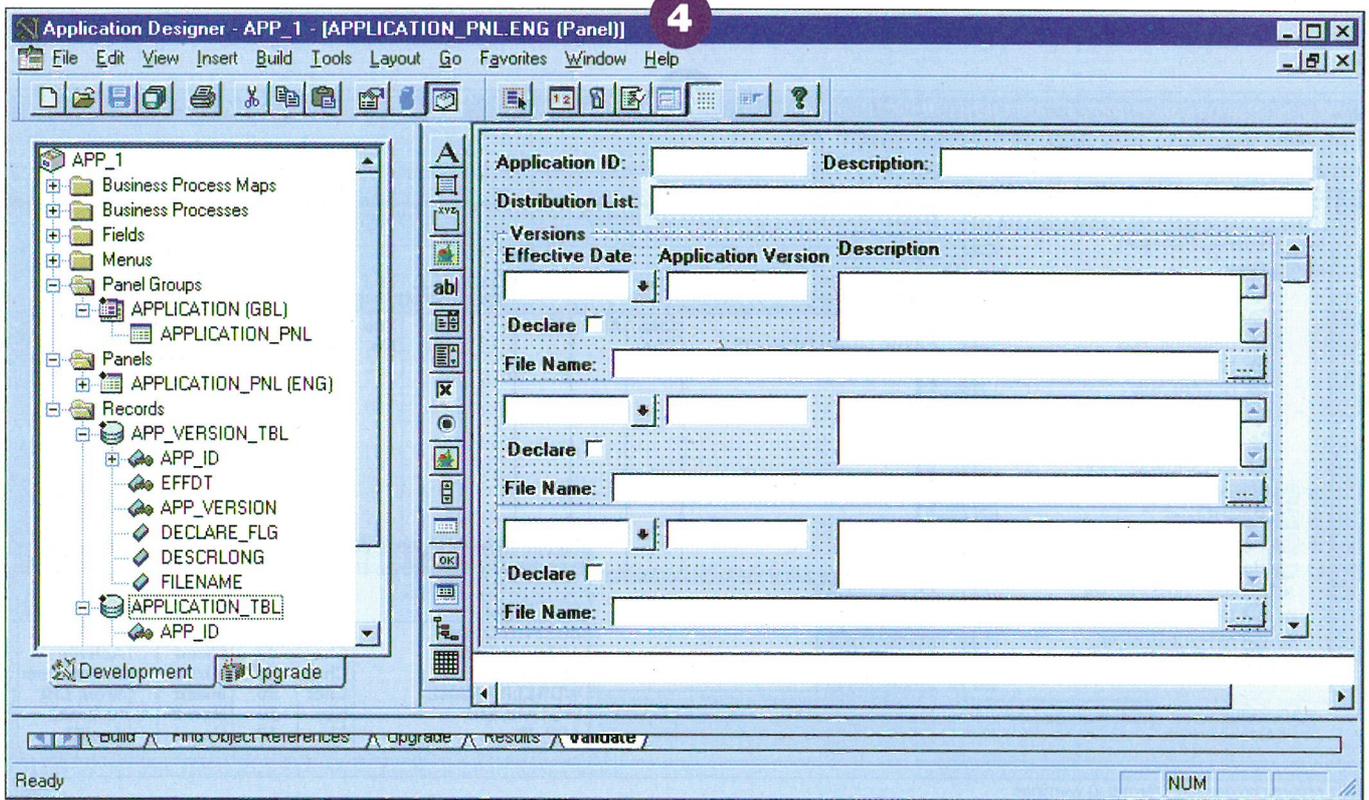
Un moteur de WorkFlow devient indispensable pour des processus de production ou collaboratifs.

En général, le WorkFlow est utilisé pour la gestion électronique de document (GED), création de dossier, routage, validation.

Voici donc un exercice pratique avec PeopleSoft 7 comme environnement de programmation, mettant en place un WorkFlow de données, donc lié directement aux tables d'une base de données SQL. Prenons, comme par hasard, la gestion de programmes et de ses versions. Chaque fois qu'une nouvelle version sort, nous aimerions automatiquement envoyer un e-mail à nos clients/utilisateurs si la version est publiable ou, dans tous les cas, placer l'information dans une liste de tâches, appelée "WorkList", pour ne pas oublier de la traiter.

## Les étapes incontournables

Les principales étapes sont la création des tables (Applications et Versions), la conception de l'écran de saisie (nommé



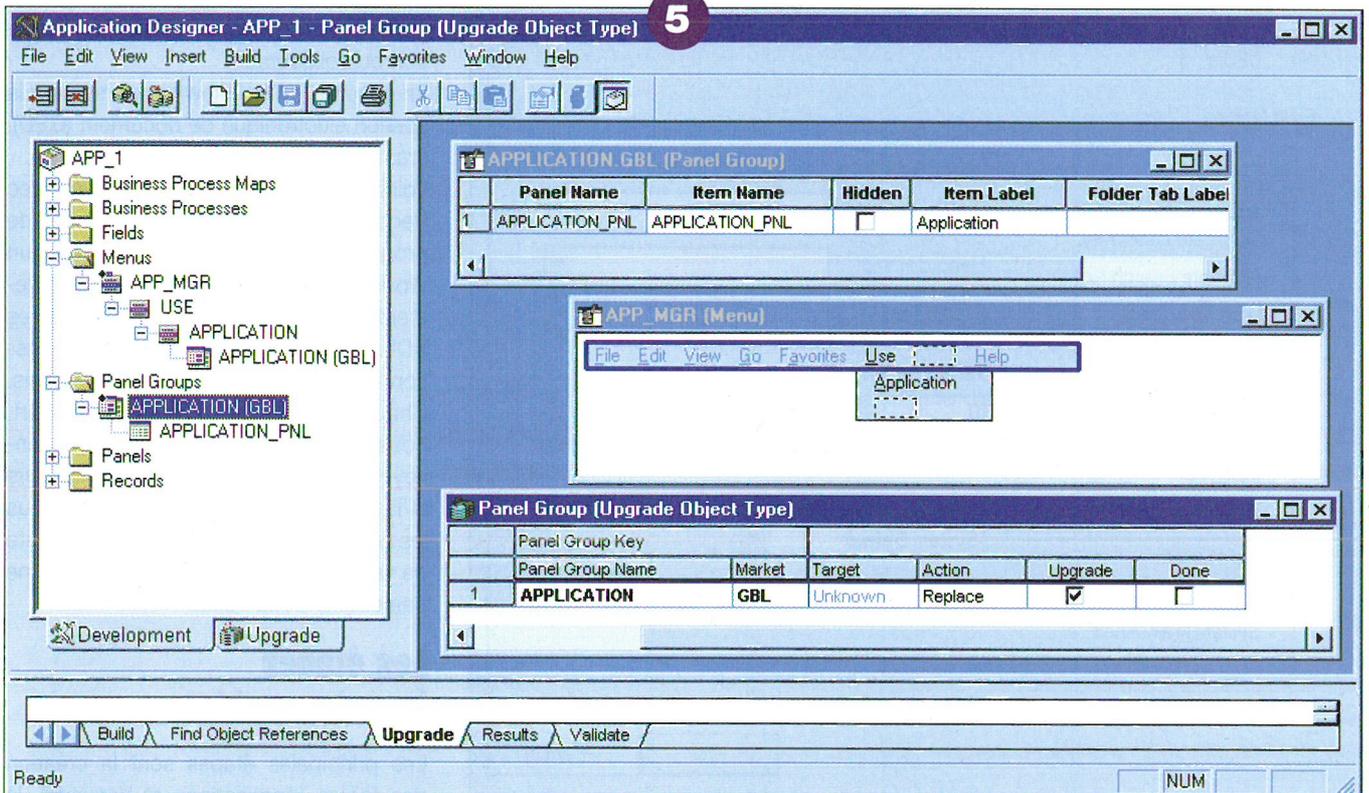
> Figure 4 : création de l'écran pour l'utilisateur final.

Panel), puis l'ajout d'un Menu. Viennent ensuite la définition des Rôles et l'assignement des utilisateurs. Enfin, on garde

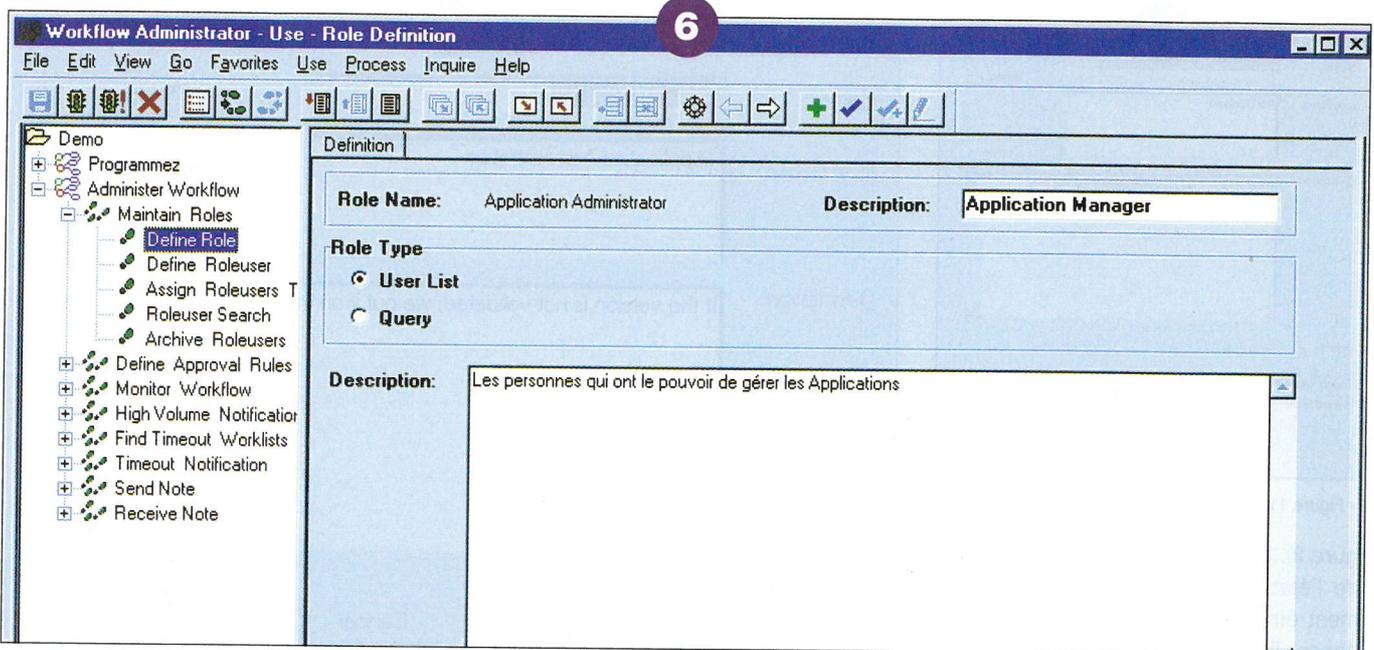
le meilleur pour la fin, avec la définition du "Business Process" comprenant les activités de votre société.

- Pour définir des tables, il faut d'abord

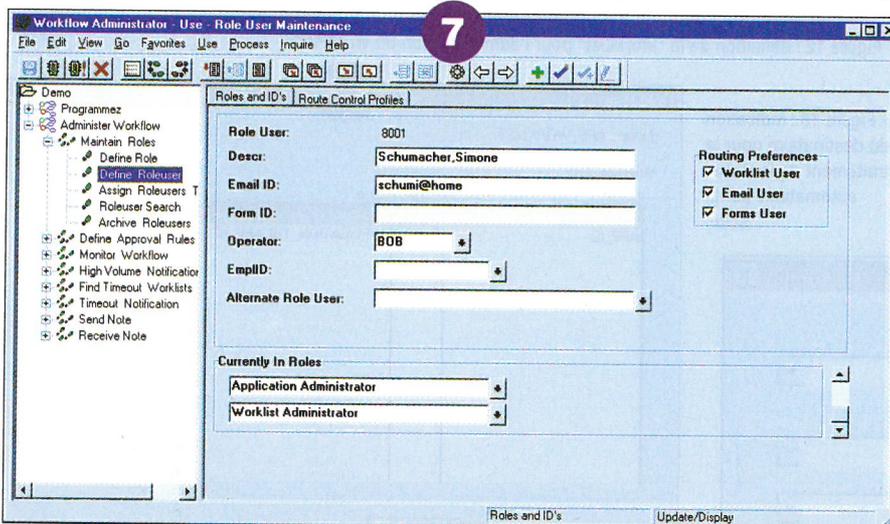
créer des champs (voir figure 1). Ensuite, en glissant ces champs dans les tables - "records" - (voir figure 2), donner les clefs et les champs de recherche (voir fi-



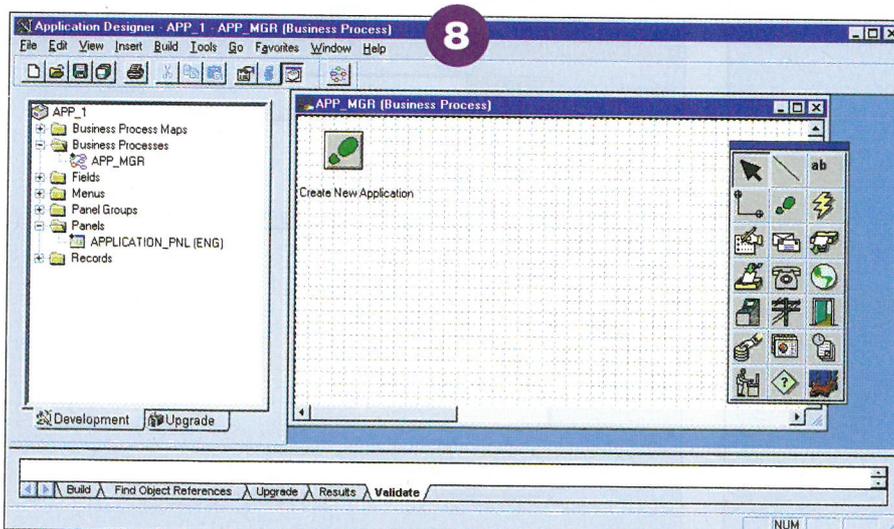
> Figure 5 : création du menu aux fonctions personnalisées pour votre application de WorkFlow.



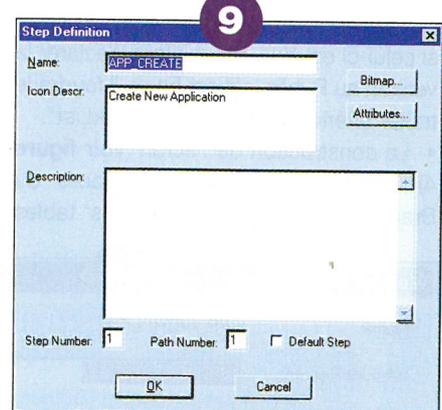
> Figure 6 : création d'un "Role" pour définir le groupe de personnes chargé de l'administration du Workflow.



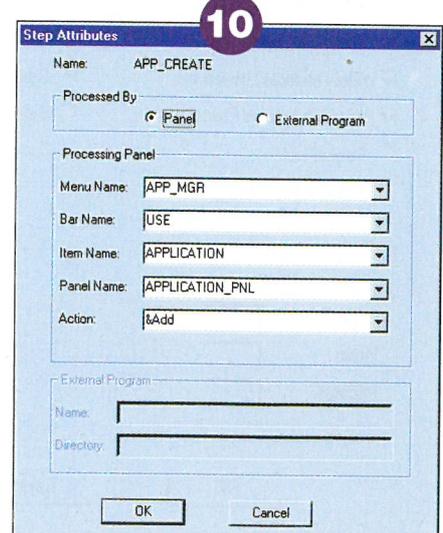
> Figure 7 : création d'un "Role User" définissant précisément la personne autorisée à l'administration.



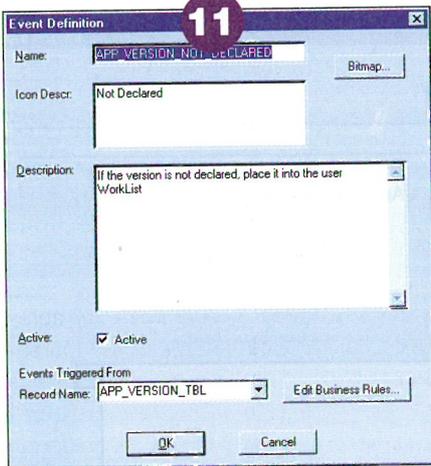
> Figure 8 : création de la première activité qui sera disponible dans l'application.



> Figure 9 : définition d'une étape dans tout le processus, ici une application nouvelle.



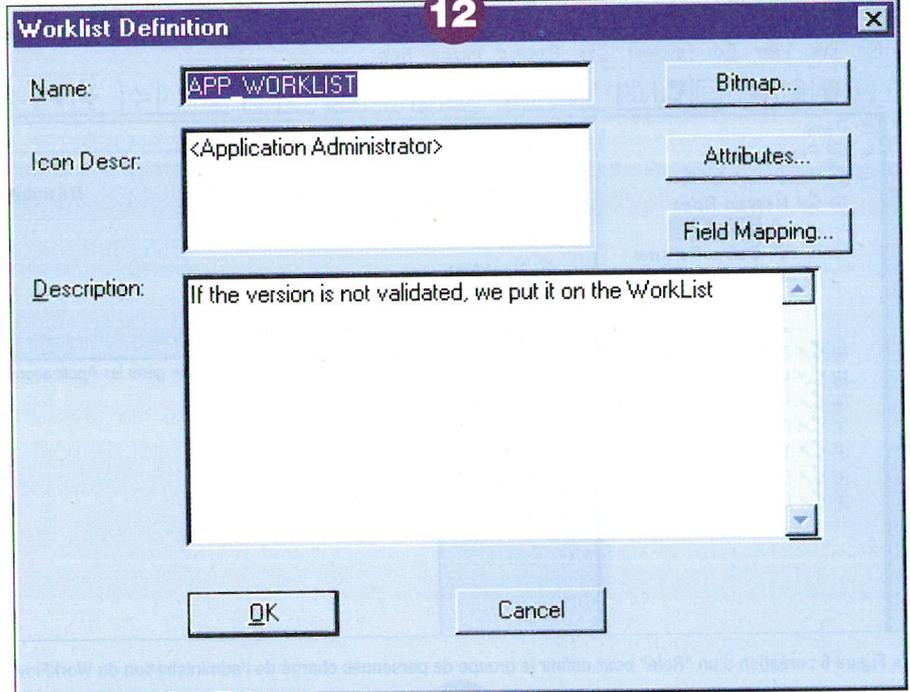
> Figure 10 : définition des attributs de l'étape.



> Figure 11 : définition d'un événement.

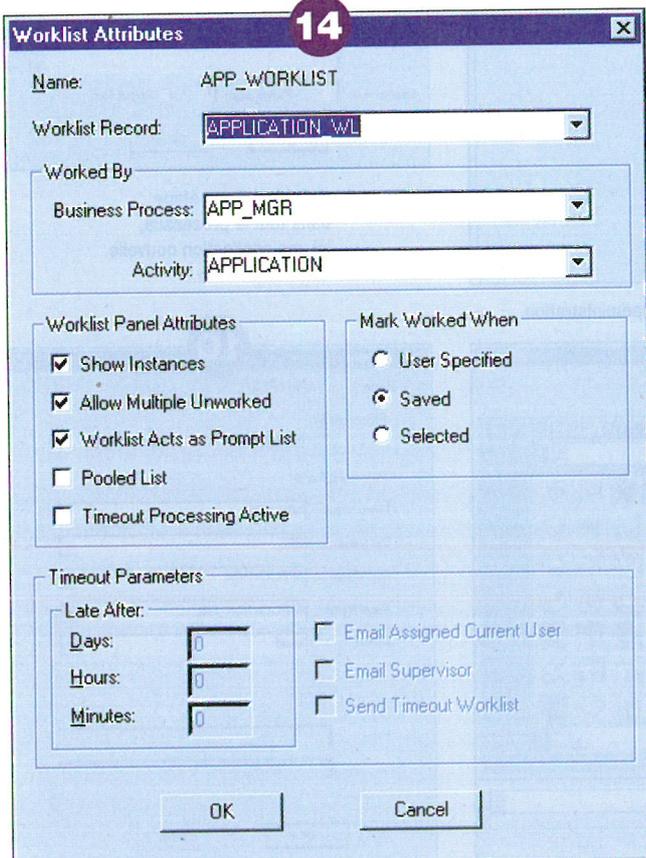
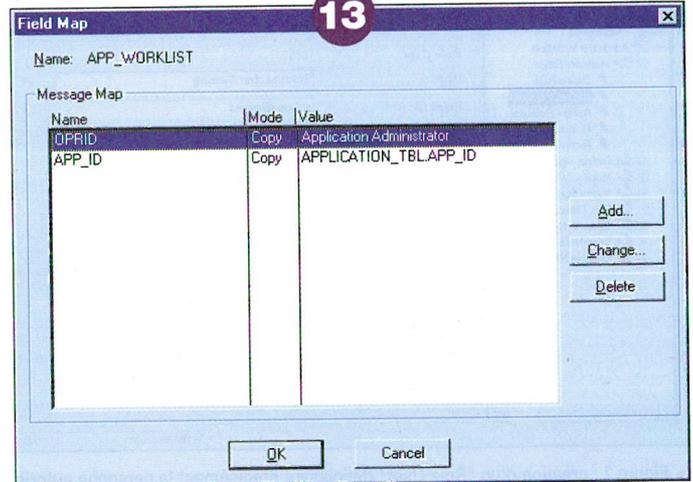
figure 3). Une table contenant les données de l'état du futur WorkFlow doit également être présente. Pour pouvoir conditionner l'envoi de l'e-mail ou pour le placer dans la "WorkList", nous ajoutons un sémaphore (flag) à la table des versions ; si celui-ci est Vrai, nous allons déclarer la version au Public, s'il est Faux, il faudra la traiter ultérieurement par la "WorkList".

- La construction de l'écran (voir figure 4) est faite en plaçant (toujours du Drag&drop) chaque champ des tables

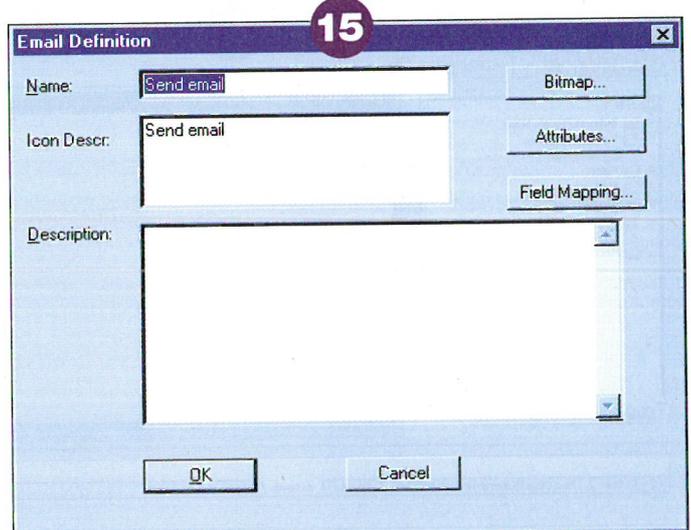


> Figure 12 : définition de la "WorkList" pour l'administration du WorkFlow.

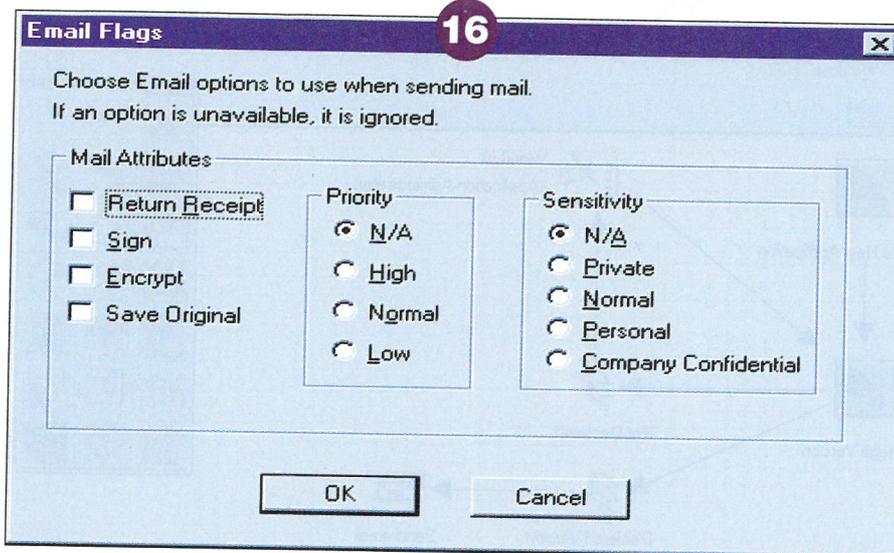
> Figure 13 : indication du destinataire pour le traitement de la tâche, automatisée par la suite.



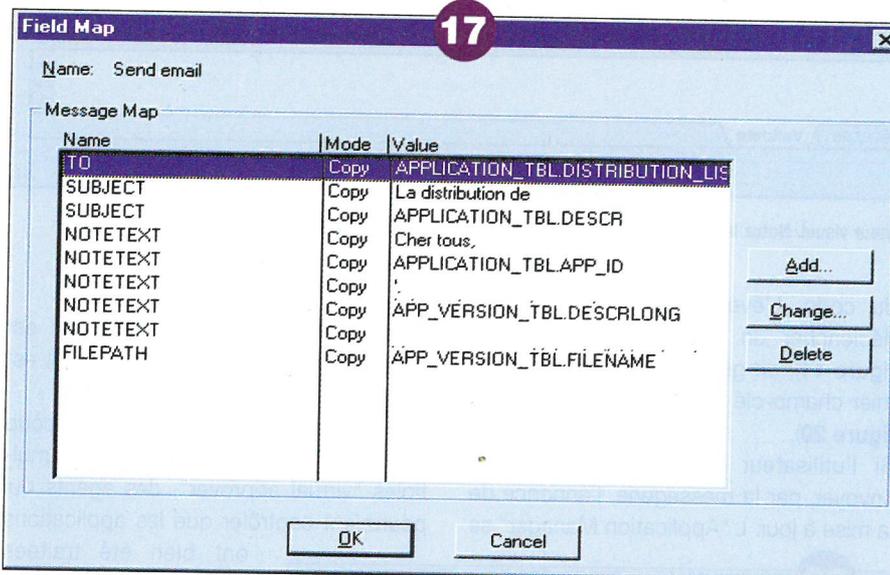
> Figure 14 : définition d'attributs pour l'exécution de l'application.



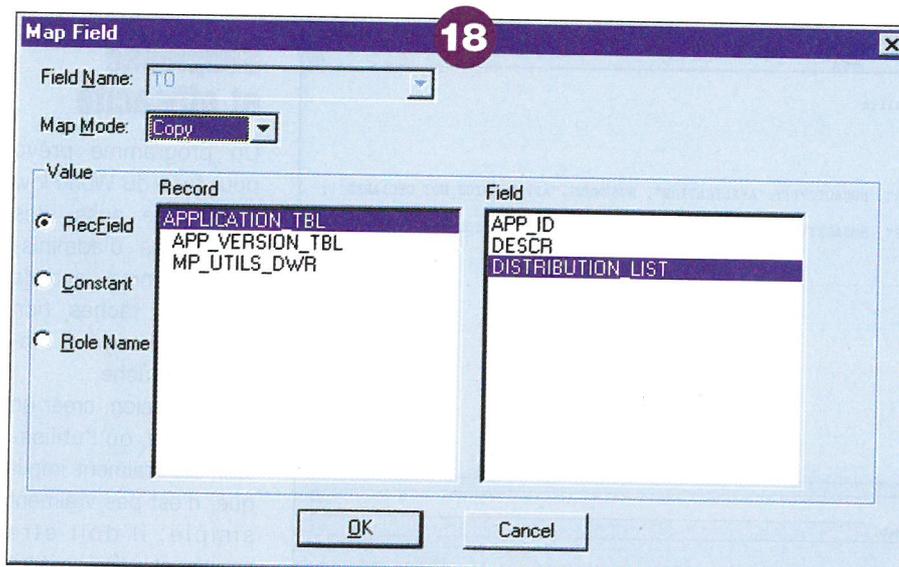
> Figure 15 : définition de l'objet e-mail indispensable pour le WorkFlow.



> Figure 16 : définition des flags de l'objet e-mail, c'est-à-dire des paramètres d'expédition.



> Figure 17 : définition des champs du message pour une meilleure exploitation.



> Figure 18 : lien entre les champs du message et les champs présents à l'écran.

dans un Panel. Les requêtes (code SQL) appropriées seront générées dynamiquement à l'exécution.

- En définissant des groupes d'écrans de saisie, on pourrait donc avoir plusieurs "Panel" accessibles par des onglets. Ici, nous n'en avons qu'un seul. Celui-ci est placé dans notre menu (voir figure 5).

- La saisie des Applications et de ces versions est d'ores et déjà possible, après avoir donné accès au Menu aux utilisateurs. Mais, comme promis, nous vous montrons un exemple de Workflow, pas uniquement la création de deux tables, d'un écran et d'un menu fait en dix minutes, sans une ligne de code. :-).

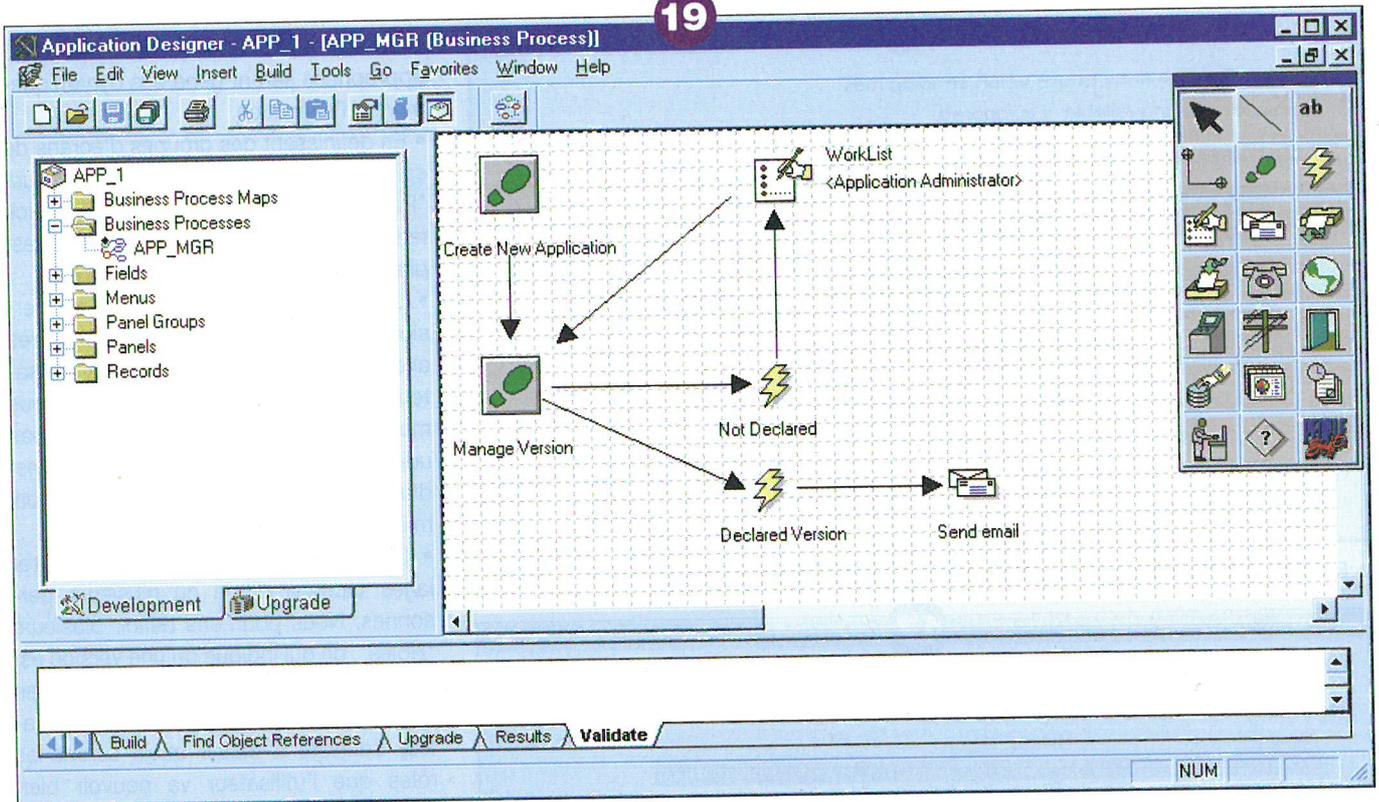
- Un "Rôle" est défini pour généraliser la/les tâche(s) d'une ou plusieurs personnes. Nous pourrions définir plusieurs "Rôles", un qui indique qu'une version est disponible et un autre qui se chargerait d'envoyer les e-mails. Le rôle de l'utilisateur consiste à définir quels seront les rôles que l'utilisateur va pouvoir bien faire. Un "Rôle" pourrait rechercher ces ouailles par un Query. Dans notre exemple (voir figure 6), nous indiquons "User List", ce qui veut dire que nous définissons les rôles de chaque utilisateur par les "Role User" (voir figure 7).

Nous ajoutons donc un "Role User" (voir figure 7) et indiquons dans "Currently In Roles" quels sont les "Rôles" que l'utilisateur va avoir. Dans notre exemple, nous mettons "Application Administrator" (que nous venons de créer) et "Worklist Administrator" qui permet entre autre de rerouter (changer) une tâche à accomplir sur une autre personne.

- Un Business Process n'est rien d'autre qu'un enchaînement de "Panels" conditionnés et d'objets divers. Si l'opérateur désire router directement cette version à la liste de distribution, il devra cocher le champ "Declare". S'il n'est pas coché, le système placera automatiquement l'information dans la "WorkList".

Au moment où l'utilisateur demandera sa liste des tâches (voir figure 19), suivant son "Role User", toutes les tâches en attente lui seront proposées. Au moment où l'utilisateur procède à une tâche, il va se retrouver dans l'écran de saisie associé à l'activité. Le code sera exécuté au moment où vous l'enregistrez sur le disque. Les événements du 'Process' se déclencheront suivant les flags respectifs

19



> Figure 19 : vue du "Business Process" terminé dans l'éditeur visuel. Notez la simplicité de modélisation.

qui leur seront attribués (voir figure 19). L'insertion de code s'impose pour déclencher l'événement approprié.

## Configurer le WorkFlow

Dans PeopleSoft, chaque champ reçoit une série d'événements pouvant exécuter

du code. L'événement WorkFlow peut déclencher un "Business Event" (voir figure 11), en général placé dans le premier champ-clé de l'enregistrement (voir figure 20).

Si l'utilisateur ne désire pas encore envoyer, par la messagerie, l'annonce de la mise à jour. L'"Application Manager" se

verra attribuer une nouvelle tâche.

Dans le cas où la case 'Declare' est cochée, le message (voir figure 23) est envoyé.

Il est clair qu'on peut faire beaucoup plus compliqué, avec des validations multiples "virtual approver", des agents qui pourraient contrôler que les applications ont bien été traitées après un certain laps de temps...

## Souplesse et efficacité

Un programme prévu pour faire du WorkFlow vous offre aussi des possibilités d'administration comme : avoir la liste des tâches non traitées, changer le statut d'une tâche...

En conclusion, créer un WorkFlow, où l'utilisateur est vraiment impliqué, n'est pas vraiment simple. Il doit être souple et efficace tout en donnant du goût.

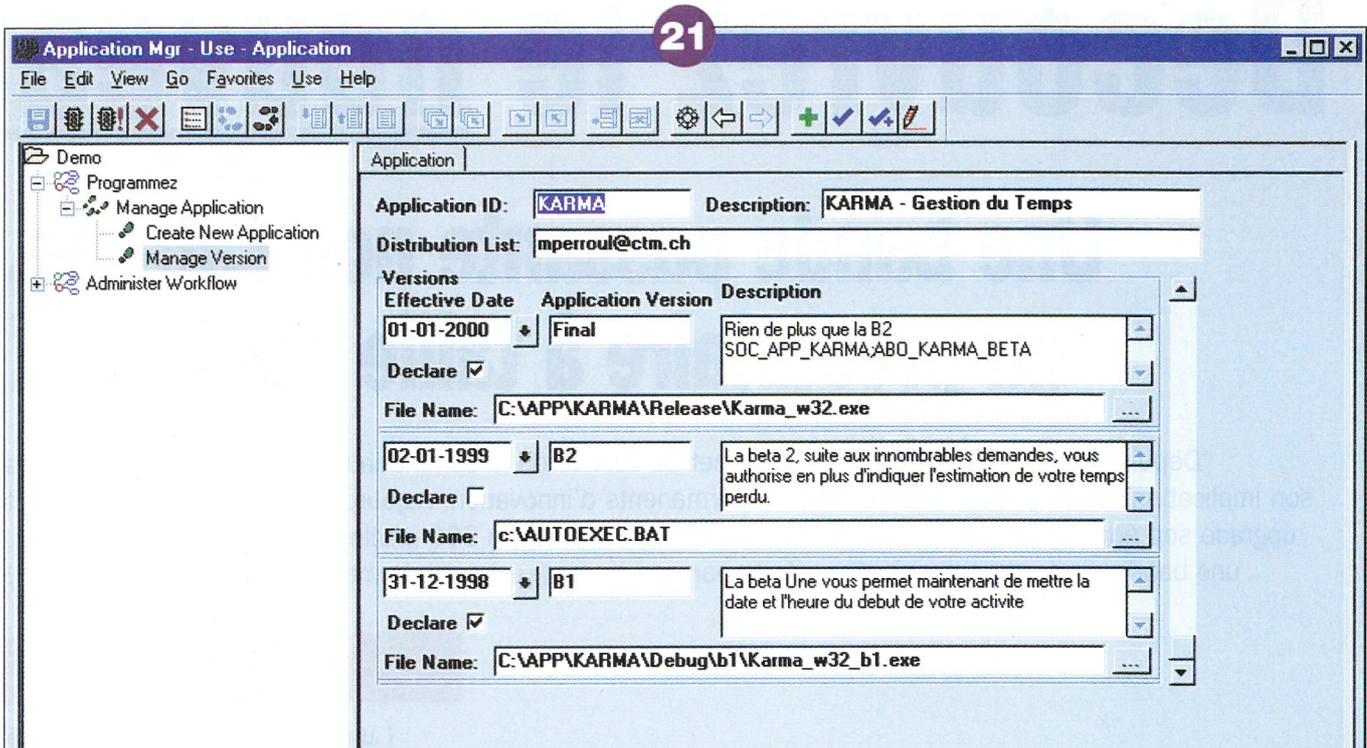
20

```

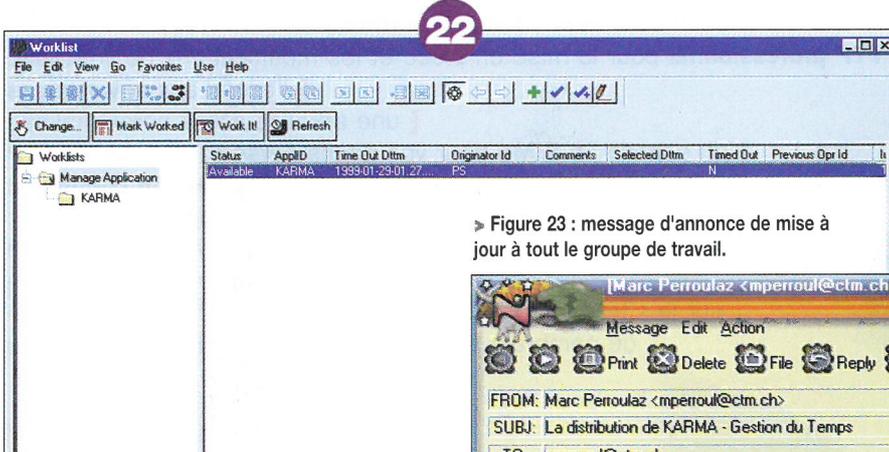
/**
 * Place dans la Worklist une version prête à être publiée
 * ou trigge l'événement de publication
 */
If FieldChanged(DECLARE_FLG) Then
  If DECLARE_FLG = "N" Then
    «BRES = TriggerBusinessEvent(BUSPROCESS."APP_MGR", BUSACTIVITY."APPLICATION", BUSEVENT."APP_VERSION_NOT_DECLARED");
  Else
    «BRES = TriggerBusinessEvent(BUSPROCESS."APP_MGR", BUSACTIVITY."APPLICATION", BUSEVENT."APP_VERSION_READY");
  End-If;
End-If;

```

> Figure 20 : voici le code de déclenchement pour activer l'application dans le groupe de travail.



> Figure 21 : création de l'application sous forme binaire et configuration détaillée de la première bêta version.



> Figure 23 : message d'annonce de mise à jour à tout le groupe de travail.

> PeopleSoft, l'éditeur du produit servant à cette démonstration.

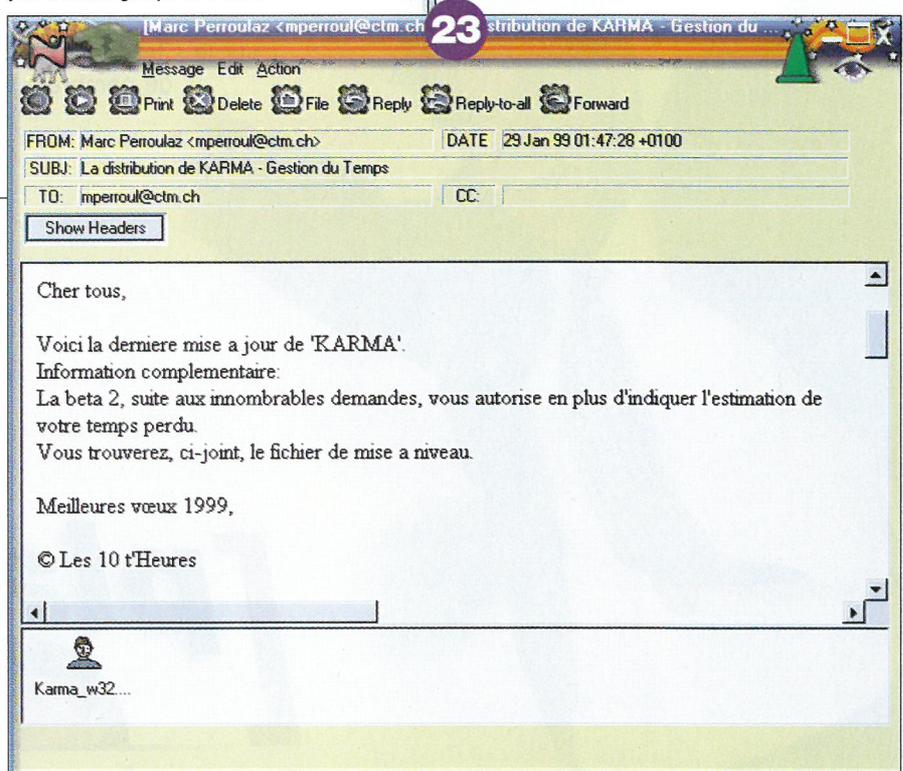


> Figure 22 : la "WorkList" avec l'application activée.

L'initiative d'en faire devrait toujours venir de l'utilisateur lui-même. Il est impératif de bien comprendre le business avant d'en faire. ■

## Quelques liens utiles

- [www.lotus.com](http://www.lotus.com) pour Lotus Notes.
- [www.csesystems.com](http://www.csesystems.com) pour CSE Systems.
- [www.peoplesoft.com](http://www.peoplesoft.com) pour PeopleSoft.
- [www.ultimus1.com](http://www.ultimus1.com) pour Ultimus.
- [www.eastmansoftware.com](http://www.eastmansoftware.com) pour WFX.



# Signe extérieur de compétence

Wall Street est, vous le savez, une terre d'exil pour certains chefs de projet français. Découvrez avec l'un d'entre eux l'opérateur égal, qui sert à identifier votre talent de codeur à travers les périlleuses possibilités d'assignation en C++.

■ par Sébastien Marc  
smarc@sailfish.com



Lorsque je dois évaluer la compétence d'un ingénieur logiciel C++, je lui demande souvent d'écrire l'opérateur d'assignation d'une classe. C'est devenu ma technique d'interview privilégiée car elle permet de jauger la compétence d'un candidat assez précisément, en posant un ensemble de questions relatives non seulement à l'opérateur mais aussi à C++ dans son ensemble.

Le but de cet article est d'apprendre à écrire l'opérateur = d'une classe et d'en comprendre les mécanismes en évoluant dans les détails de C++.

Nous allons prendre la conception d'une classe représentant une chaîne de

caractères<sup>1</sup>. Une instance de cette classe se construit à l'aide d'une chaîne de caractères :

```
class MyString
{
public:
    MyString(const char* s);
    virtual ~MyString();
private:
    char* s_;
}
```

Le constructeur est le suivant :

```
MyString::MyString(const char* s)
{
    if(s)
    {
        s_ = new char[strlen(s)+1]; //+1 à cause du caractère nul final
        strcpy(s_, s);
    }
}
```

```
} else
    s_ = null;
}
```

Le destructeur est trivial :

```
MyString::~MyString()
{
    delete[] s_;
}
```

## D'égal à égal

Je commence l'interview en demandant quels sont les défauts de cette classe. Outre l'absence de constructeur de copie, cette classe manque d'un opérateur d'assignation. Certains candidats ne pensent pas que l'opérateur soit nécessaire. Prenons le cas suivant :

```
void f()
{
    MyString s1("Sebastien");
    MyString s2("Nathalie");
    s2 = s1;
}
```

La fonction f() a toutes les chances de produire une violation d'accès sous Windows et un dump mémoire sous UNIX (voir figure 1). La raison est la suivante : la mémoire contenant la chaîne "Sebastien" est détruite deux fois. Une fois par s, une autre par s2 (voir figure 2 ci-dessous et figure 3 page suivante).

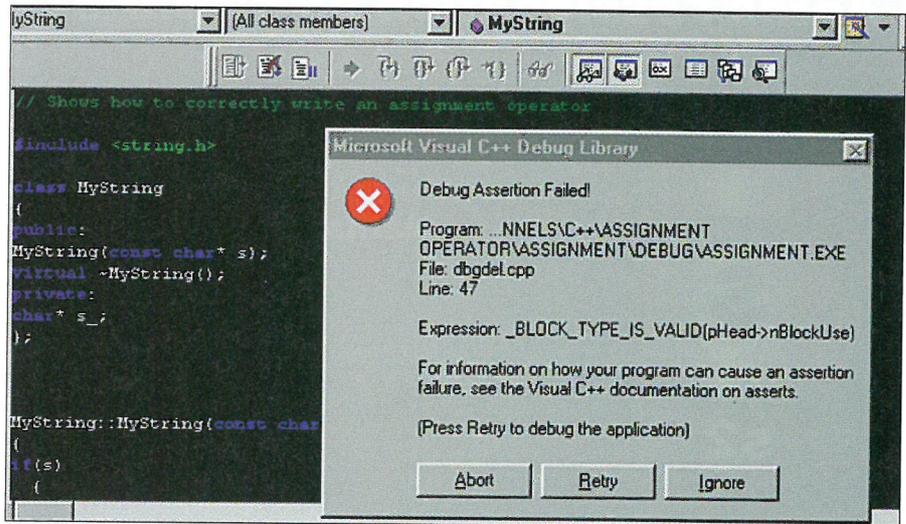
Le schéma figure 3 montre l'organisation mémoire des 2 classes après l'assignation de s2 par s1. Par défaut, le compilateur implémente une copie bit à bit des données des deux instances : les 2 pointeurs s\_ de s1 et s2 sont donc assignés à la même valeur, donc même adresse mémoire.

Premier problème rencontré : Nathalie n'est plus référencée dans notre application. C'est une fuite mémoire...

Ce n'est pas tout : Lorsque f() finit son exécution, s1 et s2 sont détruits. Le destructeur de s1 (ou s2) est appelé, libérant la mémoire occupée par "Sebastien", puis le destructeur de s2 (ou s1) est appelé, libérant la mémoire qui a déjà été libérée : c'est le crash assuré, ceci n'étant pas toléré par C++<sup>2</sup>. On en déduit une première règle : toujours définir un opérateur = lorsqu'une classe contient un pointeur<sup>3</sup>.



> Sébastien Marc est Project Leader responsable de l'implémentation de l'offre Web des produits de risk management de Sailfish Systems, filiale à 100 % de Reuters.



> Figure 1 : une violation d'accès sous Windows ou un dump mémoire sous Unix dus à la fonction f).

modifié. C'est une référence car il n'y a pas besoin de créer un nouvel objet

MyString: b existe et ne sera pas détruit durant l'assignation. La référence est plus efficace.

D'autre part, si la référence n'était pas constante, cela rendrait impossible le type de construction suivant :

```
MyString s1("Sebastien");
//...
s1="Nathalie";
```

s1="Nathalie" est interprété par le compilateur comme

s1.operator=("Nathalie");. Comme l'opérateur= ne prend pas un const char\* comme paramètre, il y a création d'un

objet temporaire, lui-même contenant la chaîne "Nathalie".

Le type des objets temporaires en C++ est toujours constant. Le type de l'objet temporaire ici serait donc const MyString&. L'opérateur doit donc prendre une référence constante, cela interdirait dans le cas contraire le type de construction ci-dessus<sup>4</sup>.

Notre deuxième règle sera la suivante : l'opérateur d'assignation prend une référence constante en argument. La valeur de retour est plus subtile. Le type de retour doit nous permettre d'enchaîner les assignations<sup>5</sup> : a=b=c. L'expression b=c doit retourner la valeur de b, donc soit un objet MyString, soit une référence sur MyString.

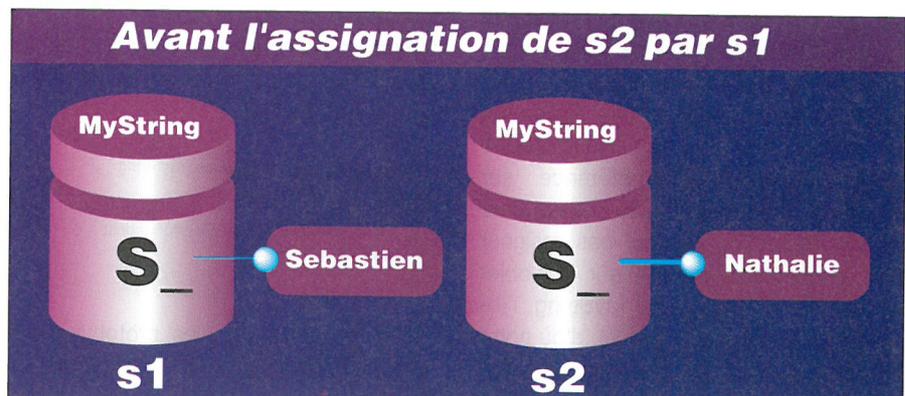
Ici également, la référence est plus efficace car nous économisons ainsi la construction d'un objet temporaire. Une des questions que je pose fréquemment aux candidats est de savoir pourquoi

## Le prototype

Si le candidat met en lumière les problèmes ci-dessus, nous commençons à implémenter l'opérateur. Le prototype est le suivant :

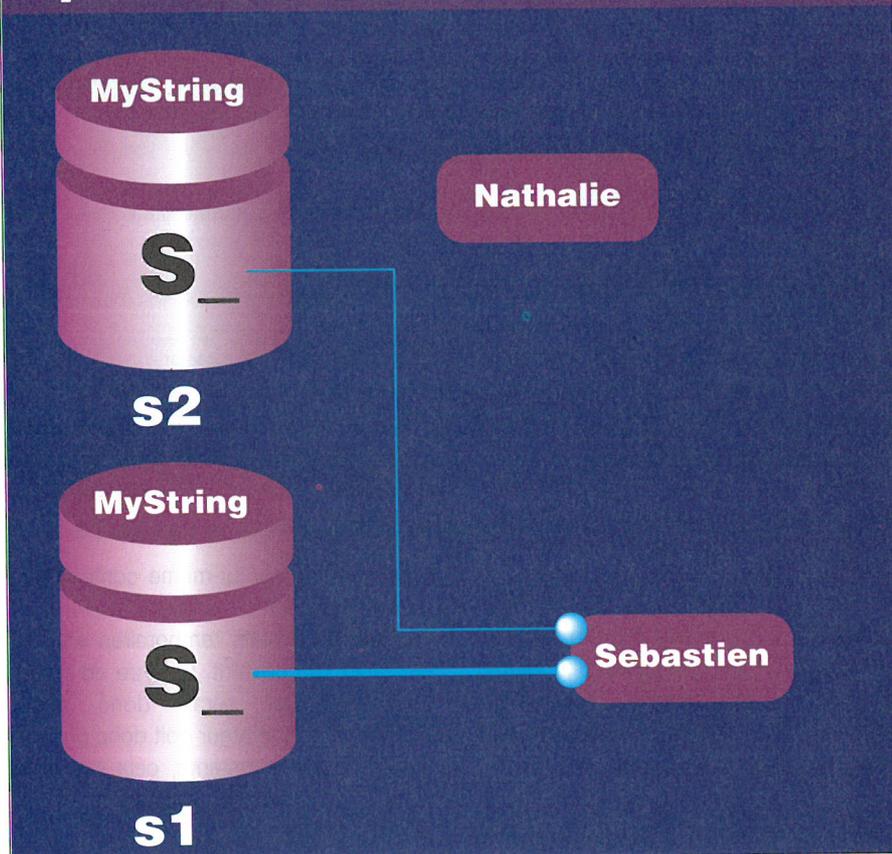
```
MyString& MyString::operator=(const MyString& copie)
```

Analysons pourquoi : le paramètre doit être constant car dans a=b; b n'est pas



> Figure 2 : tout va bien pour l'instant, ... tant qu'il n'y a pas d'assignation de valeurs.

## Après l'assignation de s2 par s1



> Figure 3 : l'objet Sébastien se fait détruire deux fois, d'où le bug identifié dans la figure 1.

nous ne pouvons pas déclarer l'opérateur de la façon suivante :

```
const MyString& MyString::operator=(const
MyString& copie)
```

La réponse tient du fait que l'on veut pouvoir permettre à l'utilisateur de faire :

```
MyString s1("Sébastien");
MyString s2("Nathalie");
MyString s3("Jean-Paul");
(s1=s2)=s3;
```

C'est autorisé par C++ pour des types simples : s1, s2 et s3 étant des entiers, le fragment de code ci-dessus assigne s2 à s1 puis s3 à s1. Etant valide pour des types simples, cela doit l'être également pour notre classe MyString. Nous devons donc retourner une référence sur MyString, autrement le code ci-dessus ne peut compiler (on ne peut pas assigner une référence constante sur un objet MyString à une référence MyString, l'inverse par contre est possible), d'où notre troisième règle : l'opérateur d'assignation retourne une référence et non pas une référence constante.

Une dernière question concernant le prototype de l'opérateur d'assignation : pourquoi ne pas le définir virtuel ?

```
virtual const MyString& MyString::operator=(const
MyString& copie).
```

Supposons que nous ayons une classe dérivée (par exemple, une classe capable d'optimiser le stockage des chaînes identiques en maintenant un compteur de référence autour du pointeur, une technique d'optimisation couramment utilisée :

```
class MyRefCountString: public MyString
{
public:
virtual MyRefCountString& operator=(const
MyRefCountString& copie);
}
```

Première remarque : les prototypes ne sont pas les mêmes, en particulier, les paramètres ne sont pas les mêmes. Le polymorphisme ne peut donc pas s'appli-

quer sur l'opérateur, C++ spécifiant qu'une opération n'est polymorphique que si les paramètres sont identiques. Supposons cependant que l'on déclare l'opérateur = de MyString de la façon suivante :

```
virtual MyRefCountString& operator=(const
MyRefCountString& copie);
```

Cette fois-ci, le prototype est identique, mais bien que l'on puisse assigner deux MyRefCountString entre elles, on ne peut plus assigner deux MyString entre elles : le compilateur refuse d'assigner une référence sur une classe de base à une référence sur une classe dérivée. Une erreur de compilation est garantie. Nous en déduisons notre quatrième règle : l'opérateur d'assignation n'est pas virtuel<sup>6</sup>.

## L'implémentation

**Premier essai :** une fois le prototype défini, passons à l'implémentation. La réponse qui vient à l'esprit est la suivante :

```
MyString& MyString::operator=(const MyString&
tocopie)
{
delete s_;
s_ = new char[strlen(tocopie.s_)+1];
strcpy(s_, tocopie.s_);
return //quelquechose du type MyString
}
```

Si vous n'avez pas sursauté en lisant cette implémentation, lisez attentivement la suite. Premièrement, le comportement de cette implémentation est indéfini. Au pire, elle fonctionnera avec votre compilateur (et

plantera sur un autre). Dans le meilleur des cas, elle plantera votre application et la plupart du temps, elle ne libérera que le premier

élément (ici le premier caractère) parmi ceux que vous avez alloués. Notre cinquième règle sera : ne jamais mélanger des types d'allocateurs/dé-allocateur mémoire. Malloc doit être libéré avec free, new avec delete et new[] avec delete[]. Notre implémentation doit être :

```
MyString& MyString::operator=(const MyString&
tocopie)
{
delete[] s_;
s_ = new char[strlen(tocopie.s_)+1];
strcpy(s_, tocopie.s_)
```

## Objet de référence sur MyString

```
return //quelque chose du type MyString
}
```

**Pouvoir s'auto-assigner** : notre opérateur a un bug qui fera planter un utilisateur qui l'utilise dans un cas bien particulier : l'auto-assignation.

Prenons l'exemple de code suivant :

```
MyString s("Sebastien");
s=s;
```

Exécutons ligne à ligne l'opérateur :

```
delete[] s_;
```

Ceci va libérer la mémoire pointée par s, le s à gauche du signe égal, mais comme le s à gauche est la même instance que le s à droite, cela aura le même effet pour le s à droite du signe égal.

L'instruction suivante va alors s'exécuter :

```
s_=new char[strlen(tocopie.s_)+1];
```

tocopie.s\_ a été désalloué par l'instruction précédente. strlen(tocopie.s\_) a alors toutes les chances de planter l'application (voir figure 4 ci-dessous).

Cette mémoire a pu être allouée pour un autre processus ou avoir été réclamée par le système d'exploitation. Toute tentative d'accès peut se révéler catastrophique. Loi de Murphy oblige, dans la plupart des

systèmes d'exploitation, cela passera inaperçu neuf fois sur dix car une autre application n'aura pas eu le temps de réclamer cette partie spécifique de la mémoire<sup>7</sup>.

Nous en déduisons notre sixième règle : toujours tester l'auto-assignation, et nous modifions donc notre opérateur de la manière suivante :

```
MyString& MyString::operator=(const MyString&
tocopie)
{
    if(this != &tocopie)
    {
        delete[] s_;
        s_=new char[strlen(tocopie.s_)+1];
        strcpy(s_, tocopie.s_)
    }
    return //quelque chose du type MyString
}
```

**Que doit-on retourner ?** Beaucoup de candidats buttent sur cette question. D'autres connaissent la réponse par cœur ou par habitude, sans vraiment savoir pourquoi. Nous savons une chose : nous devons retourner quelque chose du type MyString&. Une des solutions qui vient à l'esprit est :

```
return tocopie;
```

Essayez, cela ne compilera pas : nous essayons d'assigner une référence

constante à une référence non-constante. Cela n'est pas (et ne doit pas) être autorisé. N'essayez pas de modifier le prototype pour retourner une référence constante, souvenez-vous de notre troisième règle !

Retourner tocopie n'est donc pas une option. Quel autre choix avons-nous ? Nous savons que le pointeur this est du type MyString\*, donc \*this est du type MyString, ce qui nous convient parfaitement :

```
MyString& MyString::operator=(const MyString&
tocopie)
{
    if(this != &tocopie)
    {
        delete[] s_;
        s_=new char[strlen(tocopie.s_)+1];
        strcpy(s_, tocopie.s_)
    }
    return *this;
}
```

Nous en déduisons notre septième règle : retourner \*this pour un opérateur d'assignation.

**L'expert C++** : un candidat étant capable de donner la solution de l'opérateur tel que nous l'avons défini donne satisfaction. Pourtant, cette implémentation de l'opérateur a beaucoup de défauts.

*Toujours tester les pointeurs pour la nullité des valeurs*

Dans notre implémentation, que se passe-t-il si copie.s\_ est nul ? Nous plantons l'application.

Notre implémentation doit être modifiée de la façon suivante :

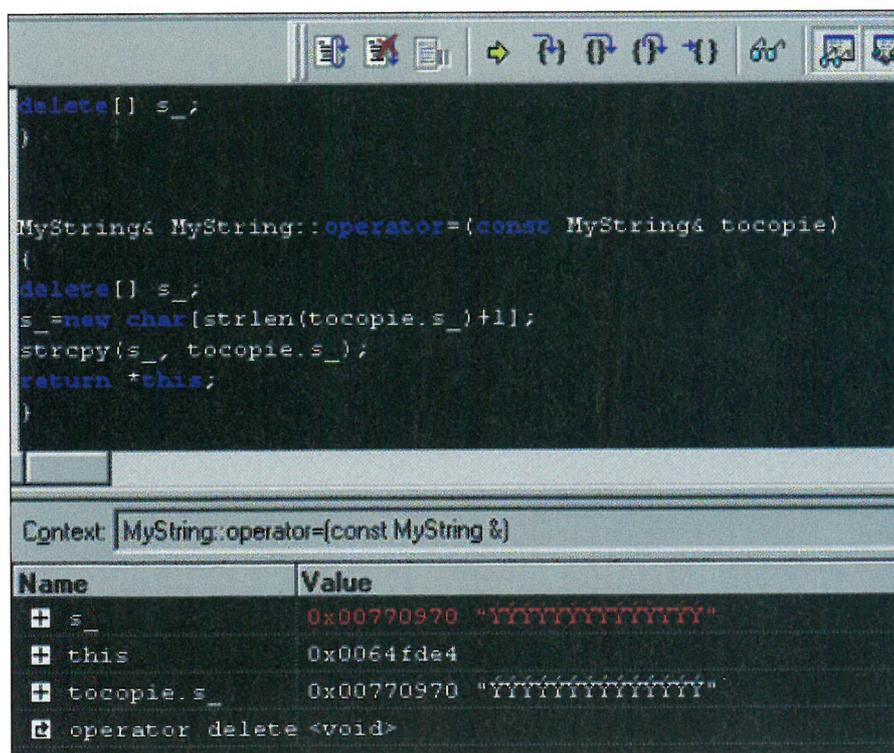
```
MyString& MyString::operator=(const MyString&
tocopie)
{
    if(this != &tocopie)
    {
        delete[] s_;
        if(tocopie.s_)
        {
            s_=new char[strlen(tocopie.s_)+1];
            strcpy(s_, tocopie.s_)
        } else
        {
            s_ = NULL;
        }
    }
    return *this;
}
```

Notre huitième règle sera la suivante : toujours tester la nullité d'un pointeur.

*Pas assez de mémoire disponible pour réaliser l'assignation*

Supposons que l'instruction s\_=new char[strlen(tocopie.s\_)+1]; ne puisse pas remplir sa tâche car la machine est incapable d'allouer un buffer de la taille de la

> Figure 4 : nous référençons une portion de mémoire invalide, en rouge sur la capture d'écran ci-dessous.



## Le code complet de la class MyString

```
#include <string.h>
#include <iostream.h>

class MyString
{
public:
    MyString(const char* s);
    virtual ~MyString();
    MyString& operator=(const MyString& s);
private:
    class StringHolder
    {
    public:
        StringHolder()
        {
            p_=NULL;
        }
        bool empty() const
        {
            return p_==NULL;
        }
        char* get() const
        {
            return p_;
        }
        operator const char*() const
        {
            return p_;
        }
        // Version simplifiée de l'opérateur.
        void operator=(char* p)
        {
            delete[] p_;
            p_=p;
        }
        ~StringHolder()
        {
            delete[] p_;
        }
    private:
        char* p_;
    };
    StringHolder s_;
};

MyString::MyString(const char* s)
{
    if(s)
    {
        s_=new char[strlen(s)+1];
        strcpy(s_.get(),s);
    }
    else
        s_=NULL;
}

MyString& MyString::operator=(const MyString& s)
{
    if(&s != this)
    {
        if(!s.s_.empty())
        {
            s_=new char[strlen(s.s_) + 1];
            strcpy(s_.get(),s.s_);
        }
        else
            s_=NULL;
    }
    return *this;
}

MyString::~MyString()
{
}

int main()
{
    MyString s1("Sebastien");
    MyString s2("Nathalie");
    MyString s3("Jean-Pierre");
    try
    {
        s1=s2;
        s3=s1;
        s1=s2=s3="Jean-Pierre";
    }
    catch(...)
    {
        cout<<"Exception catchee."<<endl;
    }
    return 0;
}
```

chaîne de caractères à copier. Les anciens compilateurs C++ retourneront NULL (et le strcpy suivant plantera l'application), d'autres respectant le standard C++ ANSI lanceront une exception. Pour la fin de cet article, nous allons supposer que nous travaillons sur un compilateur moderne<sup>8</sup>.

Supposons donc que new lance une std::bad\_alloc exception. Nous quittons l'opérateur en ayant libéré la mémoire de s\_. Imaginons la situation suivante :

```
void f()
{
```

```
s1("Sebastien");
s2("Nathalie");
s3("Jean-Pierre");
try
{
    s1=s2;
    // ...
}
catch(...)
{
    s3=s1;
}
}
```

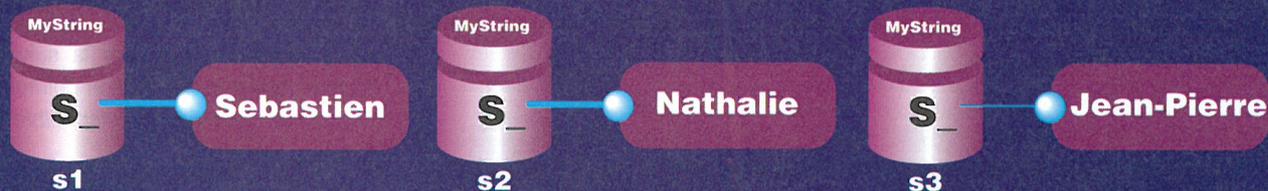
Nous créons trois objets : **voir figure 5**. Nous assignons ensuite s1 à s2, c'est ici que notre exception est lancée et que nous libérons la mémoire contenant la chaîne "Sebastien". Vous pouvez noter que la valeur du pointeur s\_ de l'instance s1 n'est pas modifiée. s1.s\_ pointe sur de la mémoire libérée.

Lorsque nous attrapons notre exception dans le bloc catch, nous assignons s1 à s3. Mentalement, exécutez le code de l'opérateur d'assignation. Nous plantons, une fois encore, l'application car dans strlen(tocopie.s\_), tocopie.s\_ pointe sur "Sebastien", mémoire qui a été libérée avant que notre exception ne soit lancée. Que faire pour éviter cette nouvelle situation ?

L'idée consiste à prendre les mesures nécessaires dans le code de l'opérateur, en attrapant l'exception dans l'opérateur lui-même. Essayons :

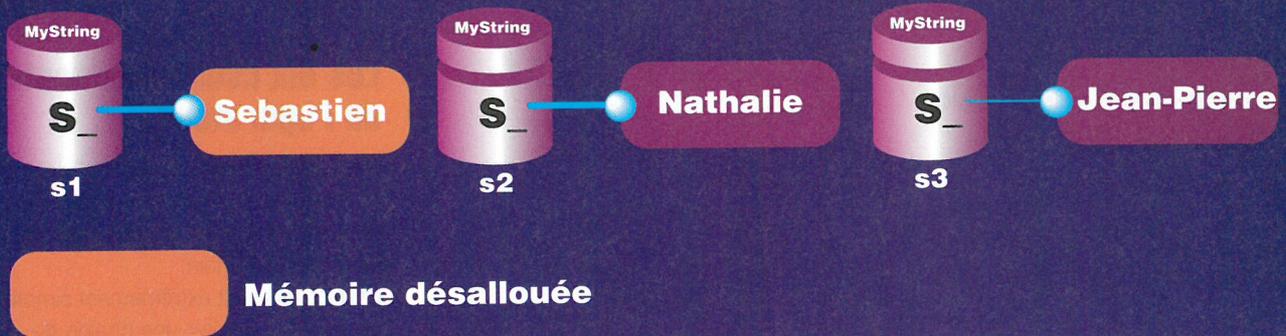
```
MyString& MyString::operator=(const MyString&
tocopie)
{
    if(this != &tocopie)
    {
        delete[] s_;
        if(tocopie.s_)
        {
            try
            {
                s_=new char[strlen(tocopie.s_)+1];
                strcpy(s_, tocopie.s_)
            }
            catch(std::bad_alloc& e)
            {
                // ...
            }
        }
    }
}
```

## Nos trois objets en mémoire



> Figure 5 : nous créons trois objets, ici les prénoms.

## Avant l'assignation de s1 à s2



> Figure 6 : il faut désallouer Sébastien, mais c'est un véritable piège tendu aux pointeurs, qui pourraient être tentés d'y accéder.

```
{
  // Que faire ?
} else
  s_ = NULL;
return *this;
}
```

Que faire dans le bloc catch ? Le comportement préconisé dans un tel cas (lorsque l'opérateur ne peut pas remplir sa fonction) est de laisser l'objet à gauche du signe égal inchangé (c'est-à-dire dans l'état qu'il avait avant l'assignation) et de relancer l'exception de sorte que l'appelant puisse savoir que quelque chose ne s'est pas déroulé comme prévu et ainsi prendre les mesures nécessaires.

Le problème ici est qu'au niveau du bloc catch, nous avons déjà affecté l'état de l'objet car nous avons libéré la mémoire occupée par s\_. Il n'y a aucun moyen de la récupérer dans le bloc catch. Nous devons donc retarder la libération de la mémoire occupée par s\_.

Une implémentation possible est la suivante :

```
MyString& MyString::operator=(const MyString&
tocopie)
{
  if(this != &tocopie)
  {
    if(tocopie.s_)
    {
      char* save = s_;
      try
      {
        s_ = new char[strlen(tocopie.s_)+1];

```

```
strcpy(s_, tocopie.s_)
delete[] save;
}
catch(std::bad_alloc& e) // catch(...) est une alternative
{
  s_ = save;
  throw; // on relance l'exception de façon à ce que
l'appelant
// puisse prendre connaissance de la situation.
}
} else
  s_ = NULL;
}
return *this;
}
```

Notre neuvième règle sera : toujours tester les conditions limites (mémoire, disque, permissions).

### Jusqu'aux limites de la mémoire

Un logiciel de qualité doit être capable de prendre en compte les situations d'erreurs, de façon à ne pas mettre en danger

l'intégrité de l'application.

Le revers de la médaille est que le traitement de ces situations peut se révéler cher, en terme de performances (ici, les try/catch/throw ne sont pas gratuits), mais également en terme de temps de développement, de ressources intellectuelles ainsi que de tests.

Une autre chose : le code tel qu'il apparaît dans le paragraphe *Toujours tester les pointeurs pour nullité* est plus lisible, donc plus maintenable que notre robuste dernière version.

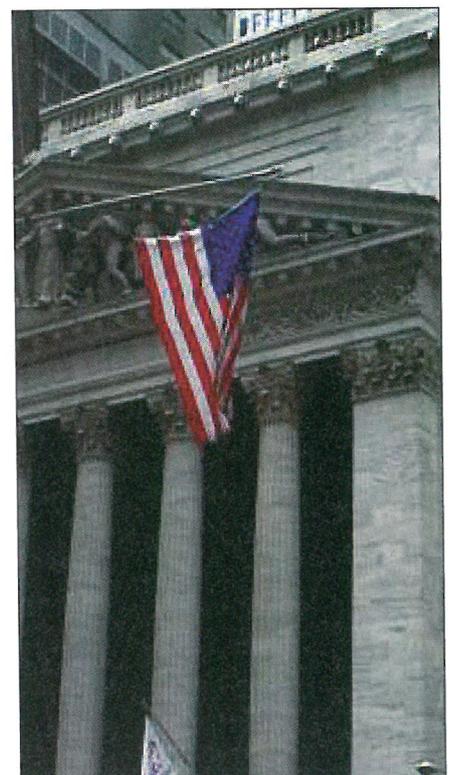
La *Standard Template Library (STL)* viendrait-elle à notre secours ?

Essayons d'adresser le problème de

maintenance. La classe `auto_ptr` de la *Standard Template Library* peut probablement nous aider puisque notre problème est relatif à l'assignation de notre pointeur `s_`. Modifions notre classe de la façon suivante :

```
class MyString
{
public:
  MyString(const char* s);
  virtual ~MyString();
private:
  auto_ptr<char> s_;
}
```

> Le siège du NYSE, le New York Stock Exchange, le décor quotidien de Sébastien Marc, project leader sans égal.



Réécrivons notre opérateur :

```
const MyString& MyString::operator=(const
MyString& s)
{
    if(&s != this)
    {
        if(s.s_.get())
        {
            s._auto_ptr<char>(new char[strlen(s.s_.get()) +
1]);
            strcpy(s.s_.get(),s.s_.get());
        } else
        {
            s._auto_ptr<char>();
        }
        return *this;
    }
}
```

C'est plus propre, les try/catch/throw ont disparu. Le contenu de s\_ ne sera libéré par l'auto-pointeur que si l'allocation s'est bien déroulée.

Si l'allocation lance une exception, le contenu de s\_ n'est pas modifié et l'exception est propagée à l'appelant, ce qui est le comportement que l'on voulait. Nous en déduisons notre dixième et dernière règle : toujours considérer l'alternative proposée par la bibliothèque Standard Template Library (STL).

## Plus propre que try/catch/throw

Il y a une erreur subtile cependant : la cinquième règle n'est plus vérifiée. En effet, l'auto-pointeur ne peut pas savoir que notre mémoire a été allouée par new[], au lieu du plus classique new. L'auto-pointeur considère que la mémoire dont il a la responsabilité a été allouée par new. Le comportement de notre dernière version de l'opérateur est donc... indéfini ! Notre version finale corrige ce problème. Nous devons pour cela introduire une sous-classe<sup>9</sup> dont le but est de palier à la déficience de l'auto-pointeur :

```
class MyString
{
public:
    MyString(const char* s);
    virtual ~MyString();
    MyString& operator=(const MyString& s);
private:
    class StringHolder
    {
public:
        StringHolder()
        {
            p_=NULL;
        }
        bool empty() const
        {
            return p_==NULL;
        }
        char* get() const
        {
```

```
return p_;
}
} operator const char*) const
{
    return p_;
}
// version simplifiée de l'opérateur
void operator=(char* p)
{
    delete[] p_;
    p_=p;
}
~StringHolder()
{
    delete[] p_;
}
private:
    char* p_;
};
StringHolder s_;
};
```

Après cette étape pleine de suspense et de String en folie, l'implémentation de notre cher opérateur devient plus complexe, mais plus propre :

```
MyString& MyString::operator=(const MyString& s)
{
    if(&s != this)
    {
        if(!s.s_.empty())
        {
            s._new char[strlen(s.s_) + 1];
            strcpy(s.s_.get(),s.s_);
        } else
        {
            s._=NULL;
        }
        return *this;
    }
}
```

Notez alors que l'implémentation de notre destructeur disparaît, puisque c'est StringHolder qui se charge de la destruction de la chaîne.

## Les dix commandements de l'opérateur

Nous avons donc dix règles à mémoriser et à respecter pour implémenter un opérateur d'assignation. Nous pouvons aller beaucoup plus loin.

Notre classe est extrêmement simple, elle ne fait que stocker une chaîne de caractères, mais elle a un comportement sain en cas d'exception. D'autres questions surviennent lors de l'implémentation d'un opérateur d'assignation d'une classe elle-même dérivée d'une autre classe ayant un opérateur d'assignation, ou lors de l'implémentation de l'opérateur d'une classe contenant des instances plus complexes qu'un simple buffer de caractères.

Le C++ est si complexe qu'il n'est pas facile de couvrir toutes ses subtilités. Cet article vous donne un aperçu réaliste des pièges à éviter. Il montre également ce qui rend le langage C++ classieux aux yeux des professionnels du développement. ■

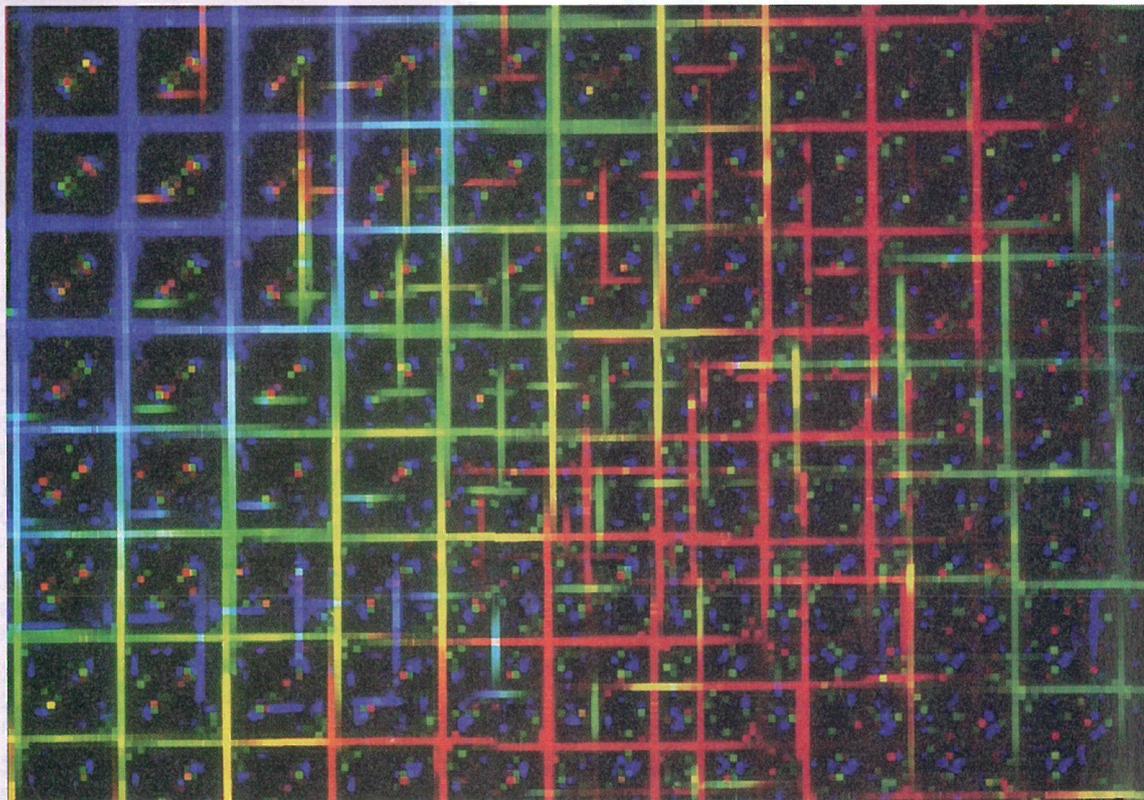
## Carnet de notes d'un exilé du C++

- 1. Ceci n'est qu'un cas d'école. Avant de concevoir une classe, vérifiez qu'elle n'existe pas dans le langage. En l'occurrence, dans notre cas, la classe string de la STL (Standard Template Library) fait parfaitement l'affaire.
- 2. Pour des raisons de performance. Il coûterait trop cher de vérifier si le bloc mémoire a déjà été désalloué ou pas.
- 3. Si la classe ne doit pas pouvoir être assignée ou si le concepteur de la classe ne souhaite pas implémenter l'opérateur, un opérateur d'assignation vide et privé doit être déclaré.
- 4. Notez que beaucoup de compilateurs C++ ont un bug à ce sujet. La création d'objets temporaires est bien souvent non constante. Cela peut avoir certains effets pervers, trop complexes pour être discutés ici.
- 5. Lorsque en C++ nous définissons un opérateur, une règle est d'implémenter l'opérateur de façon à reproduire le comportement de ce même opérateur avec des types simples (int, double...) pour ne pas surprendre l'utilisateur de cet opérateur.
- 6. Une façon d'obtenir un comportement polymorphe lors de l'assignation de deux objets est de définir une fonction clone, virtuelle, implémentée par chaque classe dérivée. Cette technique est connue sous le nom de prototype dans Design Patterns.
- 7. Des produits comme Purify ou BoundsChecker permettent de détecter ce genre de situations. Il est aujourd'hui inimaginable de mettre en production un logiciel sans avoir utilisé l'un de ces outils.
- 8. Par défaut Visual C++ (au moins jusqu'à sa version 5) retourne NULL lorsque new ne peut pas allouer suffisamment de mémoire. Il y a moyen de changer ce comportement en écrivant un gestionnaire de mémoire spécifique et en appelant la fonction set\_new\_handler en fournissant le gestionnaire personnalisé.
- 9. Inner-class en anglais. Ce terme peut vous être plus familier.

# Liste doublement chaînée générique

Si vous n'aimez pas les templates du C++, vous apprécierez cet article qui a la prétention de réinventer la roue en matière de listes génériques doubles et de pointeurs en mémoire.

■ par Claude Catonio  
dru@infonie.be



**L**a liste générique se base sur deux classes. La première est la classe Liste et la seconde la classe Maillon.

L'implémentation est assez simple. La liste possède comme propriété trois pointeurs : le pointeur vers le premier maillon de la liste, le pointeur vers le second maillon de la liste et le pointeur vers le maillon courant de la liste. Vous trouverez bien sûr toutes les méthodes classiques d'une liste : aller en début de liste, aller à la fin de la liste, aller sur le maillon suivant, aller sur le maillon précédent, obtenir le maillon courant, effacer la liste, connaître la longueur de la liste, ajouter un maillon en début de liste, ajouter un maillon

en fin de liste, ajouter un maillon dans la liste tout en la maintenant triée. Vous aurez aussi accès à d'autres méthodes que vous pouvez utiliser si vous désirez mélanger différents types de maillons dans la liste. Dans ce cas, vous pourrez vous positionner sur le premier (ou sur le dernier) maillon de ce type, aller sur le maillon suivant (ou précédent) de ce type ou encore obtenir le type du maillon courant. Vous trouverez aussi les méthodes Affiche et AfficheType qui ne sont là qu'à titre d'exemple. Vous remarquerez qu'en général, les méthodes de la liste ne font qu'appeler les méthodes adéquates du maillon concerné.

Prenons un exemple : l'insertion d'un maillon en fin de liste. L'utilisateur possède un maillon et il le passe à la méthode

InsererFin de la liste. La liste va simplement dire au maillon de s'insérer à sa fin. C'est un peu normal. Quand vous allez acheter un ticket au cinéma, vous allez vous placer de vous-même en fin de file et ce n'est pas la file qui va vous placer à sa fin. Maintenant la classe Maillon. Elle possède deux pointeurs comme propriété. Un pointeur vers le maillon qui le précède dans la liste et un pointeur vers le maillon qui le suit dans la liste. Vous y trouverez les méthodes classiques d'insertion, de suppression, de déplacement. Pour créer un maillon qui peut être inséré dans la liste, le plus simple est de penser en deux étapes. Primo, il vous faut écrire votre classe sans avoir en tête qu'elle pourra être un maillon de la liste. Par exemple :

## Astuces

■ Pour obtenir une image représentant le contenu de la fenêtre, j'ai rajouté au programme une fonction permettant de sauvegarder le contenu de la fenêtre dans un BMP.

Voici donc le code permettant, en Borland C++ Builder 3, de sauvegarder le contenu d'une fenêtre dans un fichier bitmap :

```
void __fastcall TForm1::FormDbClick(TObject
*Sender)
{
    TImage *image;

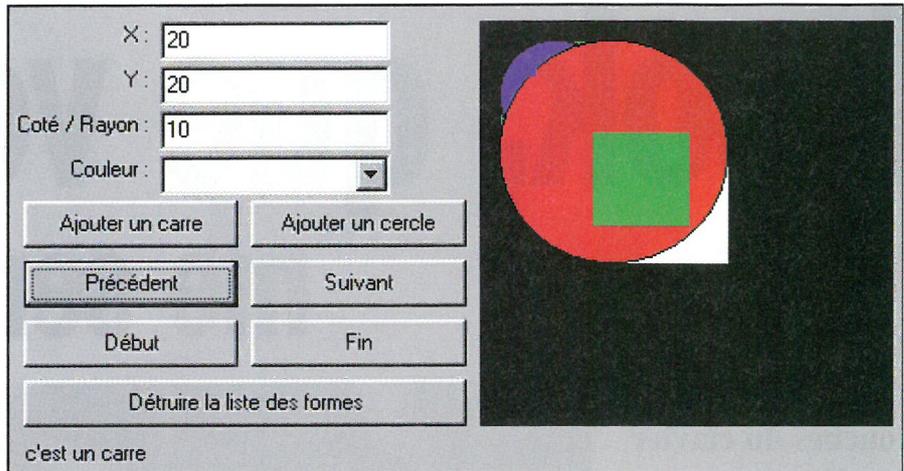
    image = new TImage(NULL);
    image->Picture->Graphic = GetFormImage();
    image->Picture->SaveToFile("image.bmp");
    delete image;
}
```

```
class entier
{
    int valeur;
public:
    void Set(int i) {valeur=i ;};
    int Get() {return valeur ;};
};
```

Une fois la classe écrite, vous pouvez vous dire que, maintenant, elle peut être un maillon. Il vous suffira alors d'hériter du maillon :

```
class entier : public Maillon
{
    int valeur;
public:
    void Set(int i) {valeur=i ;};
    int Get() {return valeur ;};
};
```

La méthode Type retourne le type du maillon. Par défaut, elle retourne type\_maillon qui est en fait l'entier 0. Elle n'a d'utilité que si vous mélangez des maillons de type différent dans la même liste. A ce moment-là, vous devez surcharger la méthode Type pour retourner une valeur qui est propre à votre classe. De même, si vous voulez pouvoir insérer les maillons en utilisant la méthode InsereTrie de Liste afin d'avoir une liste triée, vous devez surcharger l'opérateur >. La méthode Affiche du maillon est là aussi à titre indicatif. Si vous surchargez cette méthode, vous pourrez par exemple afficher l'entier (de notre exemple) à l'écran. La méthode Affiche de Liste affichera



> Le logiciel d'exemple stocke les images en liste double.

alors tous les entiers de la liste. Le programme accompagnant cet article montre des exemples de surcharge de Type et de l'opérateur >. Il s'agit d'une liste pouvant contenir des carrés et des cercles. L'ajout des carrés se fait de manière triée suivant la longueur du côté des carrés. Les cercles, eux, sont toujours ajoutés en fin de liste. Le programme est fait sous Borland C++ Builder 3.0. Le code a déjà servi sous Unix, sous DOS et sous Windows. J'ai déjà aussi créé un maillon de type

Liste. Il suffit pour cela d'hériter de Liste et de Maillon. Vous obtenez ainsi un maillon pouvant contenir une liste d'objet. On peut également créer un objet Pile, qui hérite en privé de Liste et possède les méthodes Empiler, Depiler, Hauteur, Vide, Détruire. Vous obtiendrez ainsi une pile classique. Il est aussi possible, en se basant sur la manière dont est construite la liste, de faire un arbre générique. Voilà, c'est tout. Si vous avez des questions, n'hésitez surtout pas à pilonner mon adresse e-mail. ■

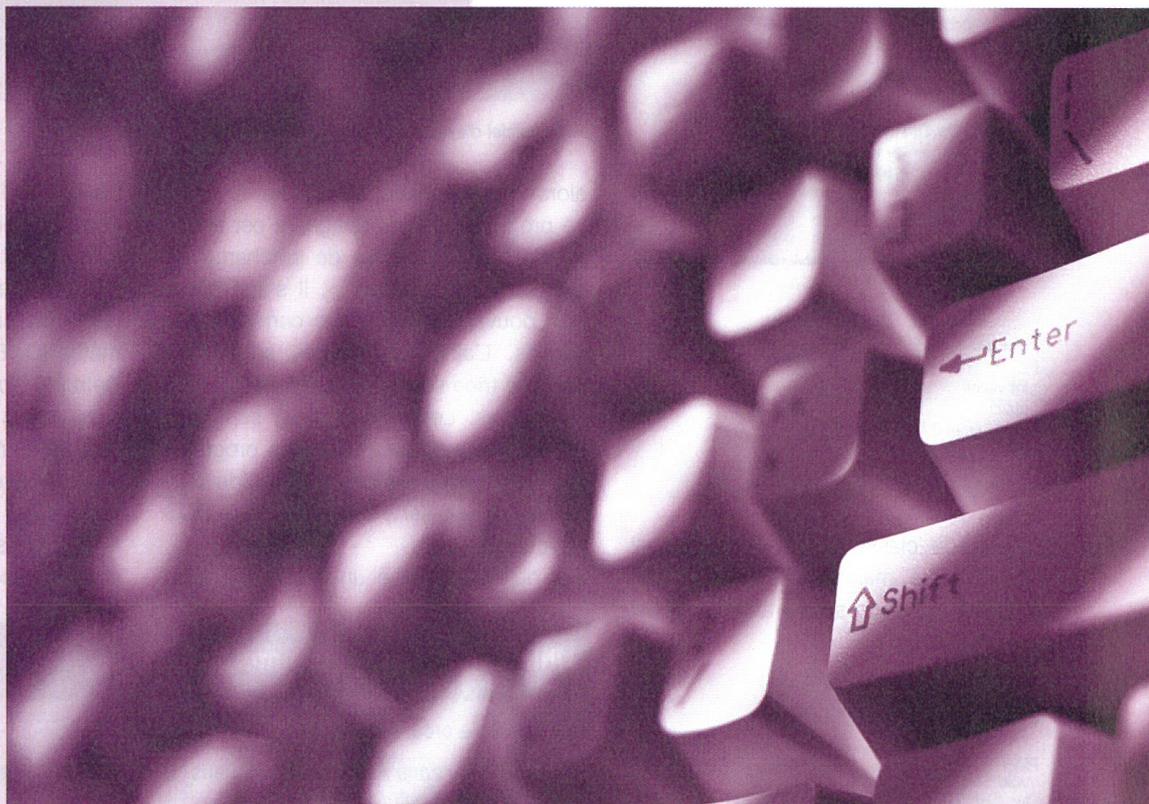
## Utilisation du programme pas à pas

- Lancer le programme.
- Rentrer 60 dans X, 60 dans Y, 50 dans "Côté / Rayon", sélectionnez la couleur vert et puis appuyez sur Ajouter un carré.
- Rentrer 30 dans X, 30 dans Y, 20 dans "Côté / Rayon", sélectionnez la couleur bleu et puis appuyez sur Ajouter un carré.
- Rentrer 50 dans X, 50 dans Y, 40 dans "Côté / Rayon", sélectionnez la couleur rouge et puis appuyez sur Ajouter un carré. Rentrer 40 dans X, 40 dans Y, 30 dans "Côté / Rayon", sélectionnez la couleur blanc, puis appuyez sur Ajouter un carré.
- Cliquez sur Début. Vous voyez alors apparaître le carré bleu. Puis cliquez sur Suivant. Vous voyez alors le carré blanc se mettre au premier plan. Cliquez sur Suivant et le carré rouge apparaît au premier plan. Cliquez sur Suivant et c'est le tour du carré vert. Cliquez sur Précédent et le carré rouge revient au premier plan. Cliquez sur Fin et seul le carré vert reste à l'écran. Bref, on voit que les carrés sont stockés dans la liste dans l'ordre croissant de la longueur de leur côté.
- Cliquez maintenant sur "Détruire la liste des formes". Et recommencez l'opération mais en ajoutant des cercles plutôt que des carrés. Vous verrez que les cercles sont stockés dans la liste dans le même ordre que celui de leur insertion (donc pas de tri).
- Maintenant, vous pouvez essayer de mélanger les cercles et les carrés dans une même liste. Détruisez la liste des formes. Insérez un carré vert de côté 50 puis un cercle rouge de rayon 40 puis un cercle bleu de rayon 20 puis un carré blanc de côté 60, un cercle vert de rayon 30 et enfin un carré bleu de côté 15. Balladez-vous dans la liste et vous verrez : les carrés sont triés et les cercles ne le sont pas. Le tout cohabite sans problème.

# Hook, que vous me semblez bon

Pour redéfinir les touches du clavier dans Windows, une seule solution : le hook. Ce crochet attrape à la volée les messages du système et offre une possibilité de traitement avant leur réception par les applications concernées.

■ par Frédéric Mazué  
frederic.mazue.wanadoo.com



**V**ous êtes accros au C, au C++ ? Vous programmez dans notre beau pays ? Si oui, vous êtes les innocentes victimes de l'ignoble clavier AZERTY. Vous en avez des crampes dans les mains et la dépression vous guette. Alors, pour ne pas accroître à vos dépens le déficit de la Sécurité sociale et en finir avec cette galère : hookez ferme !!!

Le propos de cet article est de montrer comment l'on peut, sous Windows 95/98/NT, redéfinir les touches du clavier afin de le rendre un peu plus convivial. L'exemple présenté redéfinit la touche 'imprime écran' en une accolade fermante et une accolade ouvrante, soit {} et la touche 'arrêt défilement' en un crochet ouvrant et un crochet fermant, soit [].

Les programmeurs acharnés que vous êtes n'auront donc plus à contorsionner leur jolie mimine chaque fois qu'ils délimitent une portée ou qu'ils accèdent à un élément de tableau.

## Mais comment ça marche ?

Que se passe-t-il dans les profondes ténèbres de Windows lorsque l'utilisateur presse une touche du clavier ? Tout d'abord, le hardware du clavier constitue un code appelé "scan code". Ce scan code est dépendant du matériel. C'est-à-dire que le scan code pour la touche A n'est pas le même si vous utilisez un clavier 'made by la société Dupont' ou un

clavier 'made by la société Durant'. Ensuite, le driver (pilote) de clavier se charge de traduire ce scan code en un code "virtual key" défini par Windows. A ce moment, quel que soit le fabricant de votre machine, le code virtual key est toujours le même.

Ensuite, le driver de clavier signale à Windows qu'une touche a été pressée au moyen de la routine système `keybd_event`, dont le maniement est délicat autant que capricieux. Windows est alors informé qu'un événement clavier a eu lieu et peut construire un message qui sera envoyé au processus approprié, c'est-à-dire tout simplement au processus qui détient le focus clavier au moment de l'événement.

## Newsgroup du C++ Borland

- borland.public.cppbuilder.announce
- borland.public.cppbuilder.activex
- borland.public.cppbuilder.commandlinetools
- borland.public.cppbuilder.database
- borland.public.cppbuilder.database.multi-tier
- borland.public.cppbuilder.database.sql-servers
- borland.public.cppbuilder.database.desktop
- borland.public.cppbuilder.graphics
- borland.public.cppbuilder.ide
- borland.public.cppbuilder.internet
- borland.public.cppbuilder.jobs
- borland.public.cppbuilder.language
- borland.public.cppbuilder.non-technical
- borland.public.cppbuilder.thirdpartytools
- borland.public.cppbuilder.vcl
- borland.public.cppbuilder.winapi

Enfin, la boucle des messages de la fenêtre concernée doit retirer le message de la queue et faire ce que bon lui semble du contenu du message.

Pour être complet, il reste à dire que le driver clavier envoie en général deux messages clavier pour une touche actionnée. Le premier est envoyé lorsque la touche est enfoncée, Windows construisant le message WM\_KEYDOWN, le second lorsque la touche est relâchée, Windows construisant cette fois le message WM\_KEYUP.

Bien évidemment, cela serait trop simple si cela était vrai pour toutes les touches. Il y a des exceptions, comme par exemple la touche 'imprime écran' qui ne produit un événement que lorsqu'elle est relâchée.

## L'espion qui venait du fin fond de Windows

Windows procure un moyen d'espionner tout ce trafic de messages : ce sont les hooks. Un hook est une fonction dont on a passé le pointeur à Windows. Chaque fois que dans le système survient

un événement correspondant au type de hook implanté, Windows appelle d'abord la fonction hook avant de mettre le message en queue normalement.

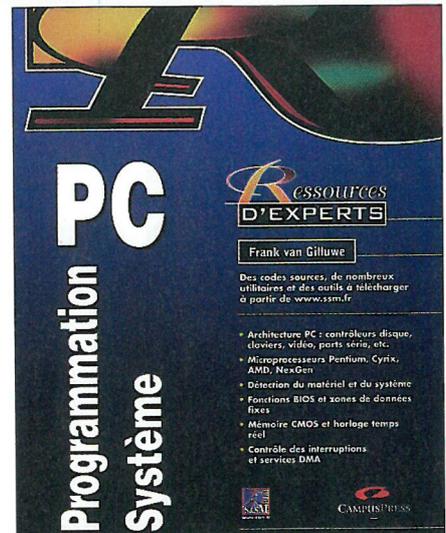
Il est possible d'implanter des hooks pour à peu près tout : procédure de fenêtre, événement souris, le shell... et bien sûr les événements clavier.

Un hook reçoit les éléments wParam et lParam des messages Windows, plus un code. Si ce code est négatif, l'utilisateur n'a pas le droit de toucher au message, sous peine de risquer de déstabiliser le système. Si le code est positif ou nul, la procédure de hook peut examiner les valeurs wParam et lParam.

## Le choix du programmeur

A ce point, plusieurs possibilités s'offrent au programmeur.

- Le message ne mérite pas d'intervention, on le laisse suivre son cours dans le système soit :
  - en retournant false,
  - en retournant le résultat de CallNextHookEx. CallNextHookEx passe les valeurs du message au hook suivant s'il existe. Quand on installe un hook, celui-ci est toujours placé en tête d'une liste de hooks. Cela signifie que si l'on est le dernier hook installé, ne pas appeler CallNextHookEx désactive tous les hooks installés par d'autres programmes. C'est



> L'ouvrage de Campus Press pour la programmation du système. 70 pages rien que sur le clavier.

parfaitement légal sous Windows, bien que pas forcément toujours bien élevé...

- Le message nécessite une intervention : on retourne true, ce qui à pour effet d'houspiller le message de la liste. Le processus à qui le message était normalement destiné ne le recevra jamais, mais on peut si on le souhaite lui envoyer un autre message en lieu et place.

Notons bien que s'il est possible d'examiner les valeurs de wParam et lParam, il est sans effet d'essayer de les modifier. Dans le cas d'un hook clavier, que contiennent ces valeurs ?

```

.. \include\winuser.h
hook.cpp  winuser.h
/*
 * SetWindowsHook() codes
 */
#define WH_MIN                (-1)
#define WH_MSGFILTER          (-1)
#define WH_JOURNALRECORD     0
#define WH_JOURNALPLAYBACK   1
#define WH_KEYBOARD          2
#define WH_GETMESSAGE        3
#define WH_CALLWNDPROC       4
#define WH_CBT                5
#define WH_SYSMSGFILTER      6
#define WH_MOUSE             7
#define WH_HARDWARE          8
#define WH_DEBUG             9
#define WH_SHELL            10
#define WH_FOREGROUNDIDLE   11
    
```

> Vous trouverez dans le fichier winuser.h les différents codes de hook.

```

.. \Include\winuser.h
hook.cpp winuser.h
/*
 * Virtual Keys, Standard Set
 */
#define VK_LBUTTON      0x01
#define VK_RBUTTON      0x02
#define VK_CANCEL       0x03
#define VK_MBUTTON      0x04      /* NOT contiguous with L & R buttons */

#define VK_BACK         0x08
#define VK_TAB          0x09

#define VK_CLEAR        0x0C
#define VK_RETURN       0x0D

#define VK_SHIFT        0x10
#define VK_CONTROL      0x11
#define VK_MENU         0x12

```

> Le fichier winuser.h contient également la définition des codes virtuels des touches.

wParam contient le code virtuel de la touche actionnée. Les valeurs de ces codes peuvent être consultées dans le fichier winuser.h ou dans le fichier d'aide de Windows.

lParam est un champ de bits constitué ainsi :

- bits 0 - 15 : valeur de délai de répétition des touches.
- bits 16 - 23 : le scan code. Oui, le scan code propre au clavier ! Que Windows transmette cette valeur après en avoir exigé la transformation à de quoi étonner...
- bit 24 : ce bit est placé lorsqu'il s'agit d'une touche "étendue". Une touche du pavé numérique par exemple.
- bits 25 - 28 : bits réservés.
- bit 29 : ce bit est placé si la touche Alt est enfoncée.
- bit 30 : ce bit est placé si la touche est enfoncée avant l'envoi du message (le cas se présente lors d'une répétition de touche).
- bit 31 : placé si la touche est relâchée, nul si la touche est enfoncée.

## Message de la Queue

Il y a deux niveaux de hook. Le niveau application et le niveau système.

Si la procédure de hook que nous avons installée est un morceau de code de notre application, le hook est automatiquement au niveau application. Cela veut dire que

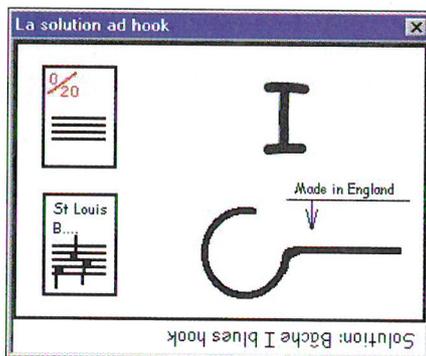
le hook est automatiquement au niveau système. La procédure de hook aura alors droit de regard sur tous les événements clavier quels qu'ils soient.

Notre but étant de définir des touches pour n'importe quel éditeur, l'écriture d'une petite dll est incontournable.

## Un droit de regard absolu sur le clavier

Rien de plus simple que d'installer un hook. Un simple appel à la fonction SetWindowsHook suffit. Cette fonction demande quatre paramètres qui sont, dans l'ordre :

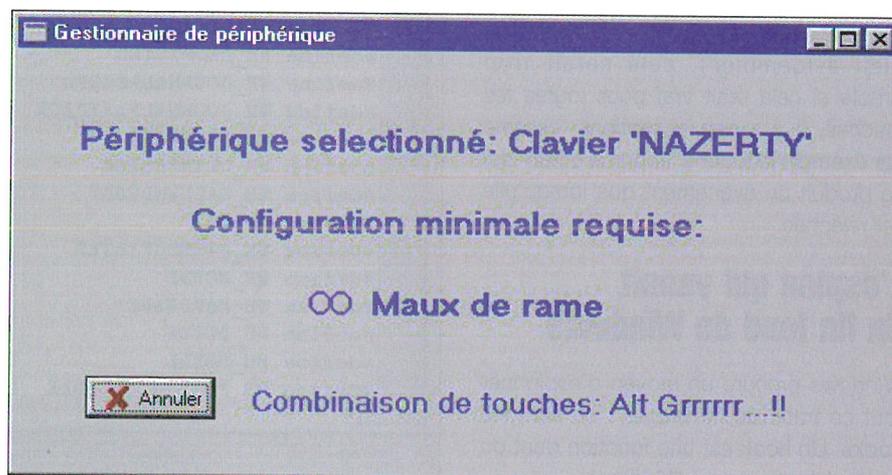
- l'identificateur du type de hook à installer. On trouvera la liste de ces identificateurs dans le fichier winuser.h ou dans le fichier d'aide de Windows. Pour notre exemple, c'est WH\_KEYBOARD ;
- un pointeur sur la procédure de hook ;
- un handle sur la dll qui contient la procédure de hook ou null si la procédure de hook fait partie du processus qui l'installe ou de l'un des ses threads (ce qui revient au même) ;
- un handle sur le thread qui contient la procédure de hook. Microsoft précise que si ce paramètre est nul, le hook sera valable pour tous les threads. Mais attention ! Mettre ce paramètre à null avec le paramètre précédant à null également ne donne pas un hook sur tous les processus. Dans ce cas, l'appel à SetWindowsHook échouera.



> Rébus de programmeur, la célèbre insulte du capitaine Hadhook dans *Tintin au pays des keyboards*.

seuls les messages qui sont destinés à notre application seront interceptés par la procédure de hook.

Si la procédure de hook est dans une dll,



> Un exemple de gestionnaire de périphérique, pour activer les touches personnalisées.

## LISTING 1 : la DLL

```
//
// Programme: hookdll.cpp, librairie dynamique
// Ecrit par F Mazué, le 02/02/1999 pour 'Programmez Magazine'
//
// Objet: Procédure de Hook clavier au niveau système qui redéfinit la
// touche 'imprime écran' en {} et la touche 'arrêt défilement' en []
// Compilé sur Borland C++Builder 3 Pro
//
#include <windows.h>
#pragma hdrstop //spécifique Borland
//
// sur un compilateur C, écrire seulement extern __declspec(dllexport) bo...
extern "C" __declspec(dllexport) bool KeyboardProc(int , WPARAM , LPARAM);
//structure message 'Maison'
struct enum_mess
{
    WPARAM wParam;
    LPARAM lParam;
};

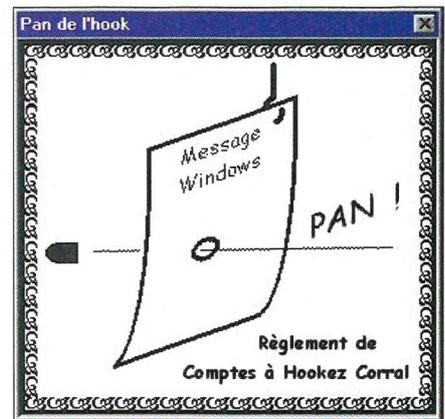
BOOL CALLBACK EnumChildProc(HWND hwnd, LPARAM lParam)
{
    struct enum_mess *mes;
    char newkey1, newkey2;

    mes = (struct enum_mess *)lParam;
    if(GetFocus() == hwnd) // si et seulement si la fenêtre a le focus
    {
        switch(mes->wParam)
        {
            case 1:
            {
                newkey1 = '{';
                newkey2 = '}';
                break;
            }
            case 2:
            {
                newkey1 = '[';
                newkey2 = ']';
                break;
            }
        }
        PostMessage(hwnd, WM_CHAR, newkey1, mes->lParam);
        PostMessage(hwnd, WM_CHAR, newkey2, mes->lParam);
        return false;
    }
    return true;
}

bool KeyboardProc(int code, WPARAM wParam, LPARAM lParam)
{
    struct enum_mess mes;

    if(code < 0) // il est interdit de toucher au message dans ce cas
        return false;
    switch(wParam)
    {
        case VK_SNAPSHOT:
        {
            mes.wParam = 1;
            break;
        }
        case VK_SCROLL:
        {
            if (lParam & 0x80000000) //si la touche est relâchée
                return true; // on coupe simplement le message de la liste
            mes.wParam = 2;
            break;
        }
        default:
            return false; // on laisse le message continuer son chemin normalement
    }
    mes.lParam = lParam;
    // on envoie le message de substitution dans la 'bonne' fenêtre.
    EnumChildWindows(GetForegroundWindow(),(WNDENUMPROC)EnumChildProc, (long)&mes);
    return true; // le message original est coupé de la liste
}

int WINAPI DllEntryPoint(HINSTANCE hinst, unsigned long reason, void*)
{
    return 1;
}
```



> Le hook n'est pas forcément d'usage bienveillant. Par exemple, il peut subtiliser le mot de passe d'une personne pendant qu'elle le saisit.

La fonction nous retourne alors un handle sur le hook créé.

Voilà notre hook installé au premier rang de la liste des hooks. De son bon vouloir seulement dépend la réception des messages clavier par les applications et même par les hooks installés avant lui.

Pour retirer le hook du système, il suffit d'appeler UnhookWindowsHookEx, avec comme unique paramètre le handle du hook créé précédemment.

## Crochet du capitaine C++

Notre visite dans les méandres de Windows s'achève pour aujourd'hui et notre clavier n'a plus de secret pour nous. Regardons le programme d'exemple. Il s'agit d'un programme écrit en C++, mais en fait dans un style C. Le but recherché étant bien de montrer un aspect du fonctionnement du système et pas de faire l'encapsulation de l'année.

L'ensemble du programme est trivial et ne nécessite guère de commentaires.

Rappelons-nous simplement qu'une action sur le clavier envoie généralement un message touche enfoncée et un message touche relâchée. Puisqu'en lieu et place on souhaite envoyer un seul message WM\_CHAR par touche, il faut retirer l'un des messages de la liste. Pour ce faire, il suffit de tester le bit 31 de lParam, ce que fait la ligne :

```
if(lParam & 0x80000000)
```



## LISTING 2 : le programme de hook sur le clavier

```
//-----
//-----
//
// Programme: hook.cpp Windows 95/98/NT
//
// Objet: Installer un hook sur le clavier au niveau système
//
// Ecrit par F Mazué, le 02/02/1999 pour 'Programmez Magazine'
//
// Signe particulier: nécessite la librairie dynamique hookdll.dll
//
// Compilé sur Borland C++Builder 3 Pro
//-----

#include <windows.h>
#include <condefs.h> //spécifique Borland
#pragma hdrstop //spécifique Borland

USELIB("hookdll.lib"); //spécifique Borland
//-----

// variables globales
HINSTANCE Instance;
HINSTANCE HookDll;
HHOOK MonHook;

// procédure de fenêtre
LRESULT CALLBACK WndProc(HWND hwnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    switch(message)
    {
        case WM_PAINT:
        {
            char *coucou = "Programmez Magazine";
            RECT rect;
            PAINTSTRUCT ps;
            SIZE size;
            BeginPaint(hwnd, &ps);
            GetClientRect(hwnd,&rect);
            SetTextAlign(ps.hdc, TA_CENTER);
            SetBkMode(ps.hdc, TRANSPARENT);
            GetTextExtentPoint32(ps.hdc, coucou, strlen(coucou), &size);
            TextOut(ps.hdc, rect.right/2, (rect.bottom-size.cy)/2, coucou, strlen(coucou));
            EndPaint(hwnd, &ps);
            break;
        }

        case WM_DESTROY:
        {
            // libération du hook
            UnhookWindowsHookEx(MonHook);
            //fermeture de la librairie dynamique
            FreeLibrary(HookDll);
            PostQuitMessage(0);
            return 0;
        }

        default:
            return DefWindowProc(hwnd, message, wParam, lParam);
    }
    return 0;
}

int PASCAL WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR, int)
{
    static char AppName[] = "Hook Programmez Magazine";
    HWND hwnd;
    MSG msg;
    WNDCLASS wndclass;
    if(!hPrevInstance)
    {
        Instance = hInstance;

        wndclass.style = CS_HREDRAWICS_VREDRAW;
        wndclass.lpszWndProc = (WNDPROC)WndProc;
        wndclass.cbClsExtra = 0;
        wndclass.cbWndExtra = 0;
        wndclass.hInstance = hInstance;
        wndclass.hIcon = LoadIcon(NULL, IDI_APPLICATION);
        wndclass.hCursor = LoadCursor(NULL, IDC_ARROW);

```

Le seul problème restant à résoudre, est de trouver la fenêtre à qui adresser le message de substitution. Certains s'étonneront peut-être de voir un appel à EnumChildWindows et non un simple appel à GetFocus pour trouver la fenêtre qui détient la focalisation. En voici la raison : les éditeurs de texte placent une fenêtre de classe EDIT dans leur zone client et cette fenêtre se charge de traiter les messages WM\_CHAR, ce qui convient à merveille à notre propos. Mais il arrive aussi que la fenêtre mère traite les messages "bruts" que sont WM\_KEYDOWN et WM\_KEYUP. Comme on a trafiqué certains messages et qu'on en a coupé d'autres, on pourrait générer un problème si par exemple une procédure de fenêtre mère un tantinet pointilleuse attend le message WM\_KEYUP censé suivre le message WM\_KEYDOWN.

Tout cela est certes improbable mais il est plus sécurisant

de s'assurer que la fenêtre qui détient la focalisation est avant tout une fenêtre fille. Vous saurez maintenant de la sorte redéfinir tout le clavier, selon vos goûts personnels ou les contraintes d'un logiciel. ■

> La dernière version du kit de développement C++ Borland.

```
wndclass.hbrBackground = (HBRUSH)COLOR_BACKGROUND;
wndclass.lpszMenuName = 0;
wndclass.lpszClassName = AppName;

RegisterClass(&wndclass);
}
else return 0;

hwnd = CreateWindow(AppName,
"Hookez ferme !",
WS_CAPTION|WS_SYSMENU,
CW_USEDEFAULT,
CW_USEDEFAULT,
200,
100,
NULL,
NULL,
hInstance,
NULL);

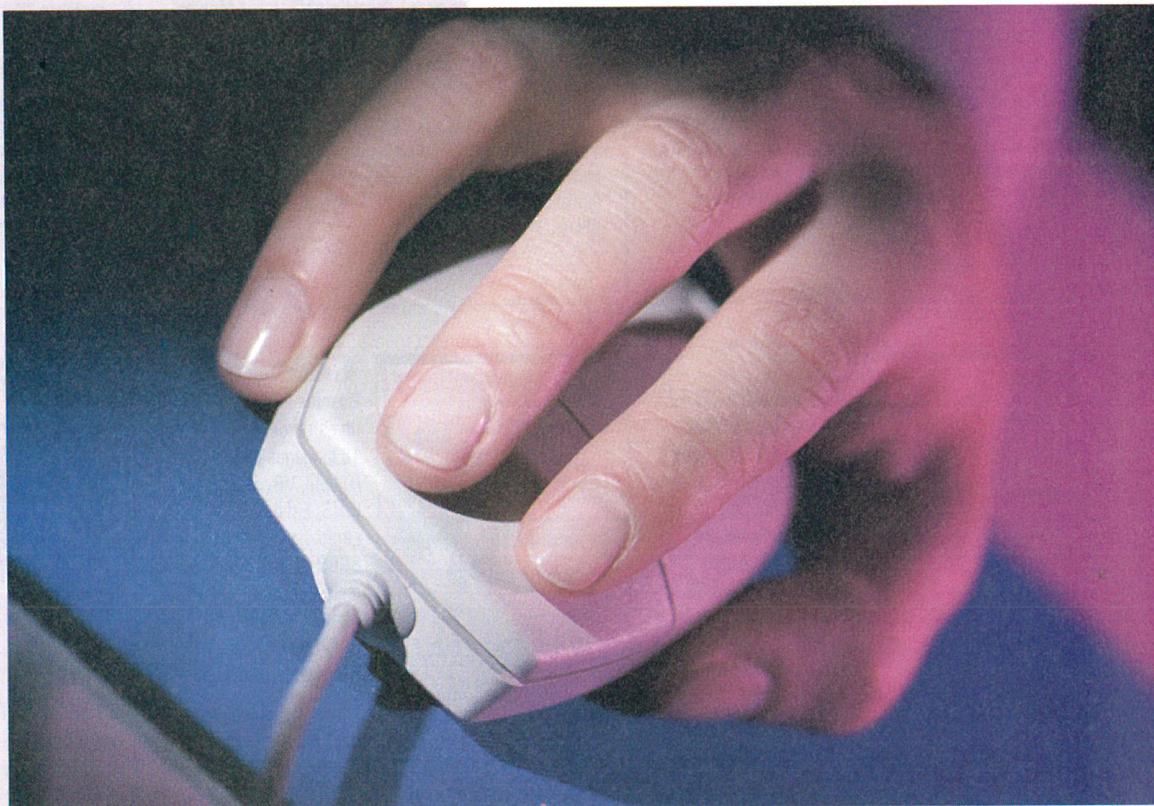
ShowWindow(hwnd, SW_MINIMIZE);
//ouverture de la librairie dynamique
HookDll = LoadLibrary("hookdll.dll");
// installation du hook
MonHook = SetWindowsHookEx(WH_KEYBOARD,
(HOOKPROC)GetProcAddress(HookDll, "KeyboardProc"), HookDll, NULL);
while(GetMessage(&msg, NULL, 0, 0))
{
    TranslateMessage(&msg);
    DispatchMessage(&msg);
}
return msg.wParam;
}
```

# Générer des aides pour le Macintosh

Des fichiers aide  
pour Macintosh ?

Quelle idée saugrenue ! Le système Mac OS est déjà le plus facile et le plus apprécié des novices. Ces documentations électroniques sont fluides et rapides d'accès. Il ne reste plus qu'à les intégrer dans vos applications...

■ Par François Tonic  
franfran@club-internet.fr



**S**ous MacOS, il existe actuellement trois grandes familles d'aide utilisateur :

- les bulles d'aide,
- le document Apple Guide,
- l'aide HTML (uniquement à partir de MacOS 8.5 et MacOS X Serveur).

Dans cet article, nous nous intéressons à la conception des aides Apple Guide qui sont les plus répandues dans le monde Mac. Pour les concevoir, l'Apple Guide SDK sera l'unique source de production. Apple Guide contient aussi bien les dernières versions des extensions système et des exemples que l'ensemble des documentations.

Mais comment concevoir un Guide ? Il existe actuellement deux méthodes :

Guide Maker et Apple Guide Visual Authoring Tool (AGVAT).

Le premier, Guide Maker, servira uniquement de débogueur et de compilateur. Par contre le second, Apple Guide Visual Authoring Tool, fournit un environnement complètement visuel d'une très grande finesse.

## Pourquoi demander de l'aide aux guides ?

Avant de vouloir concevoir un guide, il faut tout d'abord savoir comment il se structure et les technologies que l'on peut y inclure.

Un Apple Guide est avant tout un document d'aide contenant plus ou moins

d'informations. Il fonctionne avec l'extension Apple Guide. La structure d'un tel document peut paraître lourde à manipuler mais finalement, l'utilisateur navigue aisément. Quelle est cette architecture ?

- La fenêtre d'accès (ou Access Window) est une fenêtre de "démarrage" du document aide. Suivant les paramètres choisis, on aura accès aux Topics, Index et à la Recherche (Look for).
- Topics Area : on pourrait y voir la métaphore des chapitres. A chaque chapitre correspond un ou plusieurs topics. Ces topics représentent des paragraphes distincts du chapitre sélectionné.
- Les topics représentent les rubriques d'un chapitre. Ils permettent d'accéder aux panels contenant les explications.

• Un panel est un "panneau" contenant l'information par rapport au topic sélectionné.

Ainsi présentée, l'architecture ne ressemble pas à grand chose. Vous comprendrez mieux lorsque nous aborderons le développement proprement dit d'une aide.

Il faut savoir que les documents Apple Guide peuvent inclure des actions AppleScript pour une certaine assistance et interactivité avec l'utilisateur, des images (au format PICT) et même des séquences QuickTime.

## Comment concevoir un fichier d'aide

Comme nous vous l'avons précisé ci-dessus, avant d'être sous sa forme finale, un document Apple Guide est un ensemble de scripts, en fait des fichiers texte...

Tout débute avec un document Build File. C'est un fichier tout simple mais tellement important. Il permet au compilateur d'inclure tous les documents sources. Sous Apple Guide, chaque section (Content, Topic Areas, Index...) est strictement indépendante des autres. Ce qui présente un énorme avantage : la spécialisation des sources et la simplification de la maintenance.

Par exemple :

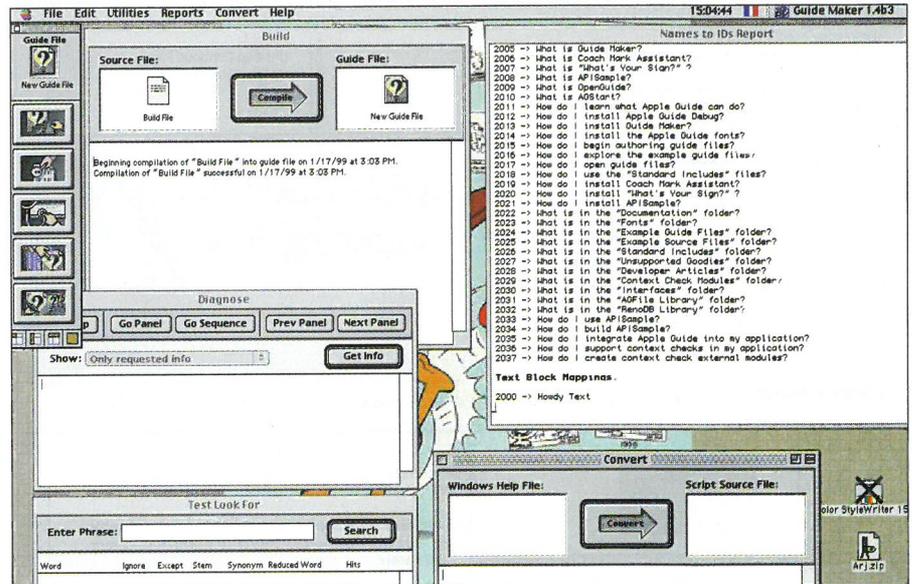
```
<Include> "Standard Setup"
<Include> "Index"

<Resource> "Standard Resources", ALL
```

La partie Resource inclut un document ressource de type ResEdit, en général pour les icones et PICT.

La seconde étape se déroule dans le Standard File. Il s'agit alors de définir toutes les options du futur document Guide. Tout est paramétrable. Par exemple, la fonction Startup Window personnalise la fenêtre par défaut d'un guide. Il ne reste plus qu'à choisir l'option : Fully, Single, avec Index, LookFor, Topics ou tout.

Si vous souhaitez un message d'introduction, pas de problème, utilisez la commande Howdy dans le Define Text Block. Comme vous pourrez le voir plus loin,



> Figure 1 : interface de Guide Maker avec quelques options pour le diagnostic, la convention et le test.

lorsque l'on travaille dans des blocs de texte, il faut définir en premier lieu le type du bloc et ensuite fermer l'ouverture du bloc, ce qui donnera quelque chose comme :

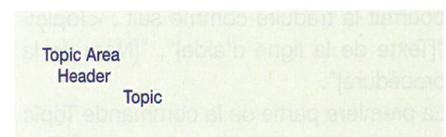
```
<Define Panel> "Installation"
Texte....
<End Panel>
```

Nous retrouvons la même chose avec les Séquences. Revenons un instant sur Standard Setup. Nous définirons les boutons avec leurs légendes, la barre de navigation, le format des Panels (type et taille), les processeurs supportés, les extensions, les ressources...

Pas d'inquiétude, cela peut paraître très lourd, mais il suffira pour débiter en douceur de partir avec un projet exemple et de la documentation. La syntaxe du Standard File est la plus complexe, avec celle pour l'inclusion des scripts AppleScript et des actions guidées (les coaches).

La syntaxe pour les panels et les autres topics est simple même s'il faut avoir le réflexe de la structure, ce que nous allons voir immédiatement.

Attaquons-nous aux Topics Area, les grands chapitres de l'aide, ceux que l'on sélectionne dans la zone de gauche de la fenêtre Guide. On peut y mettre tous les chapitres que l'on souhaite. Comment se structurent-ils ?



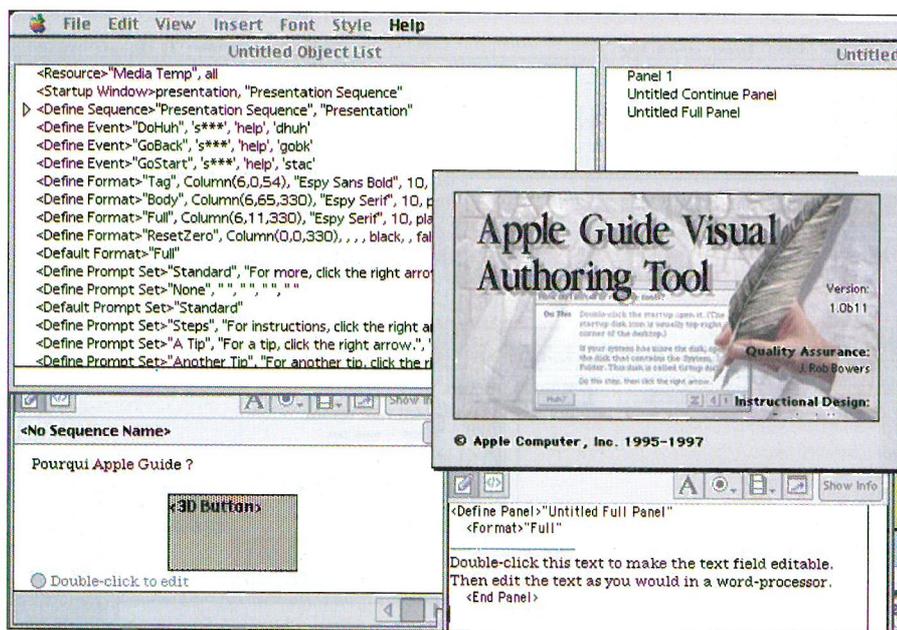
Au départ, nous avons donc un Topic Area, le chapitre. En cliquant dessus, dans la zone de droite, s'affichent les sous-chapitres avec les aides disponibles (Header / Topic).

Pour simplifier, le sous-chapitre correspond à un en-tête de section qui, en se dépliant, affiche les aides disponibles pour cet en-tête... En cliquant sur une ligne d'aide précise, que se passe-t-il ? Le Guide appelle la séquence et le panel correspondant à cette ligne. Alors s'affiche une nouvelle fenêtre avec l'explication et les boutons de navigation (retour, page suivante...).

## Les grandes lignes du code en tête

Cela vous éclaire un peu plus ? Bien, voyons maintenant le code qui se cache derrière cela, dans l'encadré qui se trouve page suivante.

Comment comprendre déjà ce premier code ? Nous en avons précédemment vu les grandes lignes plus haut. On définit dans le document Topic Area l'ensemble des chapitres, en-têtes et lignes d'aide. Vous remarquerez la syntaxe double de la commande Topic. On



> Figure 2 : vue d'ensemble de l'atelier visuel pour Apple Guide.

pourrait la traduire comme suit : `<Topic> "[Texte de la ligne d'aide]", "[Nom de la procédure]"`.

La première partie de la commande Topic donne le nom de l'aide qui va apparaître sur la fenêtre du Guide. La fin de la commande Topic fournit la "procédure" à appeler lorsque l'on clique sur cette ligne d'aide.

La procédure appelée par le Topic est une Séquence. Cette dernière inclut un ou plusieurs Panels (s'il y a plusieurs pages). En principe, ils sont contenus dans le fichier Content. Reprenons notre exemple. Nous avons appelé la séquence "Equipe", en code, l'appel se traduit donc de la manière suivante :

```
<Define Sequence> "Equipe"
<Define Panel> "Les développeurs 1"
[TEXTE]
<End Panel>
<End Sequence>
```

On déclare tout d'abord la Séquence qui correspond au contenu du Topic. Mais attention, il est nécessaire que les noms des Séquences indiquées dans les commandes Topic et Define Sequence soient exactement identiques. Dans le cas contraire, le compilateur ne pourra jamais générer le Guide.

S'il existe plusieurs Panels dans une Séquence, on déclarera plusieurs Panels dans la Séquence concernée. Comme dans toute boucle, il faut penser à déclara-

rer et à terminer les Séquences et Panels pour Define et End. Là, rien de très compliqué. Bien évidemment, on pourra créer des aides actives, modifier le format, proposer des aides complémentaires... Mais on rentre alors dans la conception lourde où la maîtrise d'Apple Guide s'avère obligatoire.

Comme il l'est précisé plus haut, rien n'interdit l'inclusion de scripts AppleScript. Cela nécessite l'intervention de plusieurs éléments : des scripts, la déclaration des appels AppleScript dans le document Content par la commande `<Coach Mark>`. Ceci réalisé, l'appel est dirigé vers le fichier Coaches qui utilise des appels du type Define AppleScript Coach pour faire correspondre le coach à un script AppleScript. Ainsi, on obtiendra du code qui ressemblera à :

```
... définition d'une Séquence et d'un Panel... (dans le fichier Content)
<Coach mark> "Ouvre le TdB"
```

```
... dans le fichier Coaches...
<DEFINE APPLESCRIPT COACH> "Ouvre le Tdb",
REDCIRCLE,":Ouvrir"
```

La commande Define AppleScript Coach fonctionne de la manière suivante : il faut préciser le nom du coach (qui est identique à celui donné dans Coach Mark). Ensuite, il s'agit de spécifier le type du cercle d'action à afficher à l'écran, dans notre cas un cercle rouge et, enfin, il faut donner le chemin et le nom du script AppleScript qui correspond.

Et encore, nous n'avons pas abordé les opérations de création d'index et de recherche... Une fois que l'ensemble des documents a été créé, encore faut-il le compiler. L'application Guide Maker est là pour cela. D'une interface simple et sobre, la prise en main de Guide Maker est rapide. On sélectionne le Build File, puis on lance la compilation. Si tout se passe bien, le document Guide est alors généré.

## Déverminer totalement les fichiers Guides

En cas de problème, la fenêtre Build listera toutes les erreurs rencontrées. Il faudra ensuite les corriger (dans les fichiers texte correspondants) et renouveler la compilation. Il est impossible de générer un guide si des bugs subsistent. On pourra tester la recherche, si cette dernière est incluse dans le Guide, mais également importer des guides Windows ou encore procéder à la localisation. Des fonctions Reports iront jusqu'à examiner les identifiants (ID) des ressources du Guide.

Avant de générer, la fonction de vérification de la syntaxe s'occupera de scanner l'ensemble du code. On peut importer des ressources ou en exporter. Le menu Appearance personnalise le titre de l'item du guide qui sera affiché dans le menu Aide... Si Guide Maker construit et teste les Guides, on ne peut que regretter de ne pouvoir les créer visuellement.

## Pour déchiffrer le code des aides

<code>&lt;Topic Area&gt; "A propos de..."</code>	-> déclaration du chapitre
<code>&lt;Header&gt; "Sur la réalisation"</code>	-> en-tête de la partie (lié au Topic Area)
<code>&lt;Topic&gt; "L'équipe", "Equipe"</code>	-> les lignes d'aide de l'en-tête
<code>&lt;Topic&gt; "La société", "Société"</code>	

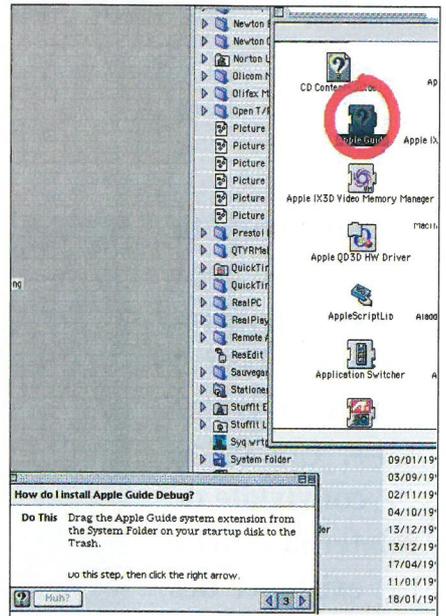
Apple Guide Visual Authoring Tool (ou AGVAT) est un environnement RAD où tout se fait visuellement : organigramme du Guide, création des Séquences et Panels, inclusion de scripts ou d'images. D'une interface riche et réussie, AGVAT comble un vide dans la technologie Apple Guide.

Revers de la médaille, si Guide Maker s'arrange de petites machines pour créer, AGVAT nécessite un Power-Macintosh (extrêmement) puissant avec une grande quantité de mémoire. Deux écrans ne seront pas superflus. En production, cet atelier se révèle puissant et rapide même si un temps de prise de main demeure indispensable. Plus besoin de passer par des fichiers textes, on manipule à l'écran

Séquences, ordre dans l'organigramme... Pour ceux qui veulent créer des guides multimédia avec séquences QuickTime et de l'AppleScript, seul Apple Guide Visual Authoring Tool les aidera à les réaliser rapidement. Mais il semble qu'Apple ne souhaite plus soutenir cet outil pourtant indispensable.

Maintenant que votre guide est fin prêt, il ne reste plus qu'à l'inclure avec votre application. N'oubliez pas lors de l'installation que le guide doit s'installer dans le dossier extension. Si l'avenir semble être la nouvelle génération d'aide HTML depuis MacOS 8.5, Apple Guide garde toujours sa place tant qu'Apple ne mettra pas à la disposition des développeurs un SDK HTML de création. Malgré tout, Apple

Guide permet de fournir des aides pour MacOS à partir de la 7.5 datant de 1994. Enfants gâtés de la micro, les macintoshiens bénéficient des meilleurs soins. Après tout, on n'est jamais trop aidé. ■



> Figure 3 : exemple d'un guide avec coach.

## Adresses Internet officielles des développeurs Apple

- [www.developer.apple.com/sdk/index.html](http://www.developer.apple.com/sdk/index.html) -> liste des SDK de MacOS.
- [www.apple.com/macos/apple\\_guide/](http://www.apple.com/macos/apple_guide/) -> site officiel de l'Apple Guide.

**HORS-SERIE COLLECTOR INTERNET COMPATIBLES**

# PCmagazine

La référence des utilisateurs avertis

**INTERNET C'EST FACILE!**

Tout pour échanger, discuter, créer

Du débutant à l'internaute confirmé

**Les MEILLEURS LOGICIELS POUR VOTRE PC SONT ICI!**

Détail au dos

**WEBCOMPASS 2.0**

COMPLÈT ILLIMITÉ

C'est un "cerveau de recherche..."

Il trouve, organise et analyse toutes les informations sur le Web

Tous les moteurs en 1 seul!

**LOGICIELS POUR INTERNET**

- 3 NAVIGATEURS
  - Internet Explorer 4.01
  - Communicator 4.5
  - Navigateur Gecko
  - Opéra (32 bits)
- 1 TRADUCTEUR DE SITES
  - Reverso/File Translator
- 1 ASPIRATEUR DE PAGES
  - MemoWeb
- LE WEB EN 10 LEÇONS
  - avec «Man Pro (en HTML)» et CD Training
- 6 ÉDITEURS HTML
  - PageMill 3.0
  - Dreamweaver 2.0 - WebExpert 2
  - Visual Page 2.0 - FileMaker Pro 4.1
  - Clear! (en HTML)
  - Flash 3.0 (en HTML) et 4.0 (en HTML)
- LES 50 MEILLEURS sharewares Internet

**EN PRATIQUE**

- Outlook Express,
- Internet Explorer 4 et 5
- Moteurs de recherche
- Les newsgroups
- L'Internet Relay Chat

**CRÉE vos pages personnelles...**

SYMANTEC

Supplément interactif de Compatibles PC Magazine Hors-Série N°29

IN 3344 - 29 - 39,00 F - RD

# Surfez efficace !

Enfin une découverte d'Internet par la pratique au travers des questions que vous vous posez !

- la sélection des meilleurs sites pratiques et de loisirs
- le point sur la sécurité sur Internet
- tout pour maîtriser Outlook Express
- comment mieux naviguer avec Internet Explorer 4
- les nouveautés d'Explorer 5
- quels moteurs de recherche choisir
- comment fréquenter les NewsGroups
- créer ses premières pages Web
- les équipements pour mieux se connecter (utilitaires, modems, caméras, livres)...

**EXCLUSIF !**

Offert sur le CD-Rom :

**WEBCOMPASS 2.0** de Symantec

Un système de recherche sous Internet en version complète et illimitée

# Le Jeu de la vie

## orientée objet

Voici un petit exercice pour novices, montrant l'art de vivre et de survivre en C++. Une façon conviviale de relater la cruelle existence des objets.

■ Nicolas et  
Jean-Bernard Boichat  
jean@tell.ascom.ch,  
a.clutch@worldcom.ch



Nous nous sommes rappelé du "Jeu de la vie" que l'on retrouve par exemple dans des mises en veille de stations de travail de Sun Microsystems. Cependant, cet article parle avant tout du langage C++ et de la manière de développer et tester une application.

Le Jeu de la vie a été inventé par J. Conway et présenté dans *Scientific American* en octobre 1970. Ses règles sont les suivantes :

- Une cellule continue à vivre si elle a deux ou trois cellules voisines.
- Une cellule meurt d'isolement si elle a moins de deux voisines.
- Une cellule meurt d'étouffement si elle a plus de trois voisines.

- Une cellule naît dans une case vide, si trois cases voisines exactement sont occupées.

Voici quelques exemples représentant l'évolution des générations : voir figure 1 page suivante.

### La classe Vie

Les listings 1 (cf. CD-ROM Vie.h) et 2 (cf. CD-ROM Vie.cpp) représentent le code de la classe Vie qui permet l'implémentation des fonctions nécessaires au Jeu de la vie. Le programme du listing 3 (cf. CD-ROM Viemain.cpp) permet d'une manière simple de tester la classe Vie. La compilation devrait pouvoir se faire sans difficulté sur différentes machines

ou environnement (PC ou Unix).

Je ne décrirai ici que les parties essentielles. La méthode prochaine() calcule la génération suivante avec les règles définies par J. Conway. Pour ce faire, la variable compteur est utilisée avec le tableau position[8][2] qui nous permet d'identifier la location des 8 voisins.

Le tableau jeu[][] est utilisé d'abord comme copie de calcul et ensuite pour sauvegarder l'ancienne génération. Cela nous permettra plus tard d'analyser les générations vieillissantes et naissantes.

La classe Vie génère un tableau dynamique pour la grandeur du monde. Il aurait été trop simpliste de définir un monde fixe (par exemple un jeu[80][25] pour un écran avec 25 lignes de 80 caractères).

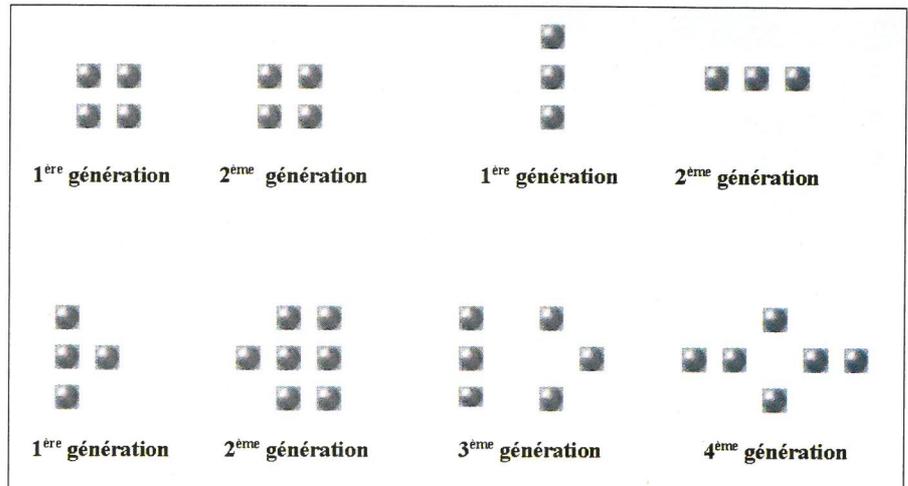
tères. La méthode initialise() alloue la table dynamiquement. La manière de faire est sans doute la partie de code la plus intéressante de ce programme.

Le monde est un rectangle auquel j'ai ajouté des bords à deux dimensions pour simuler un monde infini. Mais ce n'est pas 100 % correct puisque des objets peuvent se déplacer et dans mon cas tomberaient dans une faille !

**Exercice 1 :** il s'agit de modifier le monde pour qu'il soit fini et fermé. Les voisins des bords sont de l'autre côté et les \*\*\* doivent alterner.

La méthode au\_hasard() est triviale et la lecture du fichier dans Viemain.cpp permet de tester le code avec la méthode dessine(). La description de la méthode dimension() est décrite dans le code : elle retourne une valeur progressive pour animer la croissance ou décroissance du martien (utilisé plus tard dans la partie directDraw).

La classe Vie utilise un minimum de fonctions C. On peut se débrouiller très bien avec les bibliothèques standard du C++.



> Figure 1 : exemple d'évolution aléatoire des générations de cellules dans le "Jeu de la vie".

Personnellement j'aime bien les istream(buffer) >> largeur qui ont été mes premières découvertes en passant de C à C++.

J'ai toujours affirmé que s'il ne devait me rester qu'un seul outil en C++, je garderais celui-là. On trouvera plus d'informations sur ce produit commercial en

consultant le site Web [www.rational.com](http://www.rational.com). Prenons par exemple ce morceau de code :

```
char jeu[0][8];

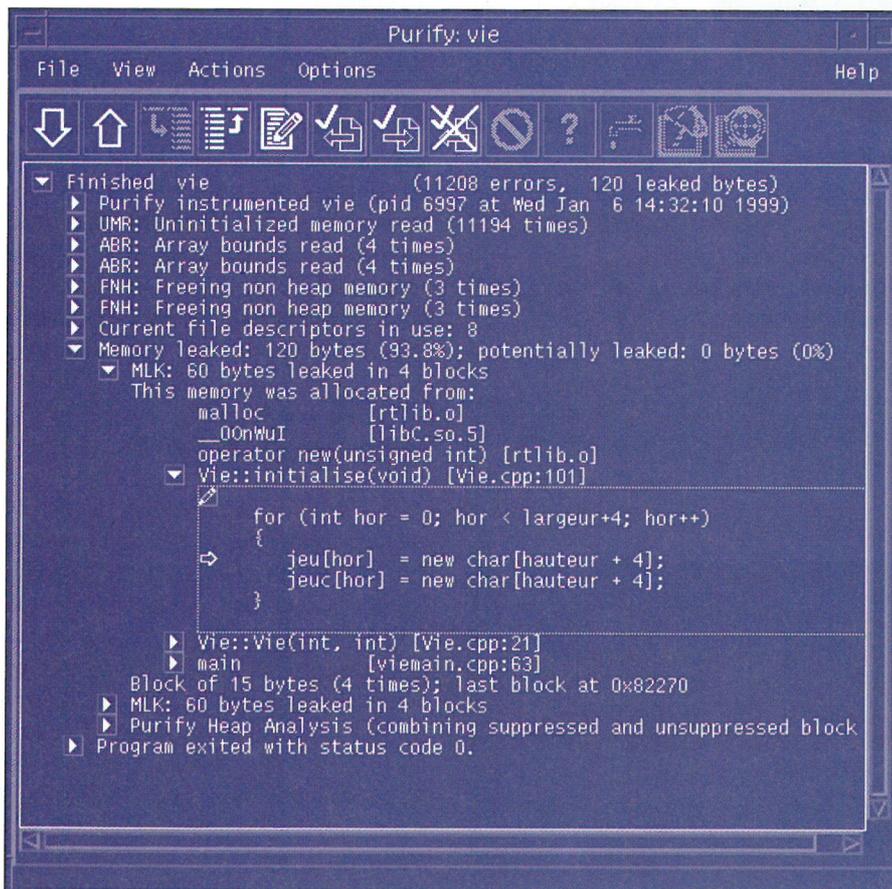
jeu[0][8] = 1;
if (jeu[0][7] == 0) { ...};
```

La première faute est évidente, l'index de jeu[0][8] n'est pas possible (0 à 7). La deuxième est beaucoup moins évidente. La position [0][7] existe mais n'a pas été initialisée. Suivant le type de fautes et de compilateur et suivant la structure du programme, on devra sans doute attendre plusieurs jours ou semaines avant que le programme fasse une faute (pourquoi pas le 01/01/2000 où il y aura des erreurs de logiciel qui n'auront rien à voir avec le nouveau millénaire).

## Purifier les objets

Purify est un excellent outil qui nous permet de découvrir ce type d'erreur pendant l'exécution ou à la terminaison du programme. Voici un exemple d'une fenêtre de Purify alors que je développais la classe Vie et que j'avais encore quelques erreurs. Les fuites mémoires (memory leaks) peuvent être catastrophiques pour des processus permanents utilisant et libérant une multitude d'instances. En ayant par exemple :

```
for (int hor = 0; hor < largeur + 4; hor++) {
    delete[] jeu[hor + 4];
}
```



> Figure 2 : le développement de la classe Vie, commence par Finished Vie dans l'outil Purify.

## Tableau 1

- ESC - Quitte
- F1 - Pour l'aide
- F2 - 50 générations
- F8 - Réinitialisation

// au lieu de :

```
for (int hor = 0; hor < largeur + 4; hor++) {
    delete[] jeu[hor + 4];
    delete[] jeu[hor + 4];
}
```

On voit bien comment le processus pourrait utiliser toujours plus de mémoire (puisque qu'on oublie `jeu[][]`). Je me souviens d'un processus Unix qui avait atteint plus de 50 Mbytes de mémoire, ralentissant toujours plus le système, avant de se planter. Et ceci avait pris plus de deux mois ! Bien que dans mon entreprise tout code doit être "purifié" avant d'être livré pour les tests d'intégration, j'ai moi-même l'habitude d'utiliser Purify dès le début du développement. Il y a deux raisons à cela :

- lorsque le programme est terminé, il y a souvent trop d'erreurs et cela prend du temps pour s'y retrouver,
- une petite erreur de code peut faire apparaître un gros problème de design. Purify génère le code binaire au niveau du link et la compilation peut se faire ainsi :

```
purify CC -o Viemain Viemain.o Vie.o
```

Dans l'encadré "Purify 1 - pendant le développement" on trouve les termes sui-

vants : voir tableau 1.

Le dernier encadré "Purify 2 - en fin de développement", c'est on ne peut plus parfait !

## La classe Traceur

Les listings 4 (cf. CD-ROM Traceur.h) et 5 (cf. CD-ROM Traceur.cpp) représentent le code de la classe Traceur qui permet de suivre le déroulement des parties critiques du programme. C'est le programmeur qui décidera où les utiliser. On pourra bien évidemment réutiliser cette jolie classe pour d'autres applications DOS ou Unix. Le constructeur reçoit un nom de fichier qu'il crée et ouvre.

Les méthodes `write()` reçoivent des paramètres définis par le programmeur et suivant l'information à disposition. Elles écrivent dans le fichier une trace de l'endroit où il passe avec des données correspondantes.

**Exercice 2 :** il s'agit d'étendre la classe Traceur en y ajoutant la date et l'heure. Le code présenté étant utilisable sous

DOS ou Unix, il est évident que cette extension entraînera des restrictions (C++ n'est pas Java).

Plusieurs objets Traceur peuvent coexister. Par exemple :

```
int main(int argc, char **argv)
{
    Traceur tmain("main.txt");
    tmain.write("Main started"); // trace dans main.txt

    if (i < ...)
    {
        Traceur tif("if.txt");
    }
}
```

```
tif.write("if executed. i = ", i); // trace dans if.txt
tmain.write("if executed"); // trace dans
main.txt
} //tif destructeur
} //tmain destructeur
```

**Exercice 3 :** il faut réécrire la classe Traceur avec :

- a) un seul fichier pour toutes les instances,
- b) une identification pour chaque objet Traceur (ex : Traceur tmain("main")),
- c) chaque ligne commence avec la date, l'heure et l'identification,
- d) le fichier de trace est conservé,
- e) Traceur actif avec une variable d'environnement ou un fichier traceur.ini.

## POV-RAY Tracing

Il ne faut pas manquer le superbe site Web de Persistence of Vision (POV) : [www.povray.org/](http://www.povray.org/). On y trouve la documentation ainsi que les outils pour différentes machines. L'outil pour Windows est très bien fait : on y édite les scripts, on les exécute et on constate le résultat. POV est un langage qui nous permet de créer des objets dans un espace à trois dimensions.

Pour le programme DirectX qui va suivre, il nous fallait une image et on a trouvé intéressant d'utiliser POV pour fabriquer notre martien. Le listing martien.pov nous montre comment construire notre objet 3D composé de sphères et d'un cylindre. Le résultat produit a été ensuite traité par Paint Shop Pro enfin d'obtenir une image assez claire ainsi qu'un fond vert pur correspond au masque nécessaire à DirectX. Voici le résultat : voir figure 3.

## Le Jeu de la Vie avec DirectDraw

Le livre de Microsoft Press *Atelier DirectX* est nécessaire pour les débutants et pour tirer parti au maximum de cette technologie. Le CD qui l'accompagne contient le SDK 5.2 de DirectX. Pour ceux qui possèdent uniquement le SDK ou même les dernières versions 6.0 ou 6.1, ils peuvent également consulter les exemples `ddex*` dans le répertoire `sdk\samples`. Le site Web de Microsoft se trouve sur [www.microsoft.com/directx](http://www.microsoft.com/directx). cf. le tableau ci-contre.

Le listing `VieX.cpp` qui utilise les classes Vie et Traceur décrites dans cet article est



> Figure 3 : un martien Pov-Ray figolé en bitmap.

## Tableau 2

Code	Signification
UMR	Uninitialized Memory Read. Lecture de mémoire non initialisée.
ABR	Array Bound Read. Lecture en dehors d'un bloc mémoire alloué.
FNH	Freeing Non Heap Memory. Libération de mémoire qui ne se trouve pas sur le heap (le stack par exemple).
MLK	Mémoire perdue. Pas de pointeurs à ce bloc ou à l'intérieur de celui-ci.

## Terminologie DirectX

- **Surface primaire** : surface affichée à l'utilisateur.
- **Surfaces off-screen** : tampons de stockage qui viendront ensuite sur la surface primaire ou la surface basculante.
- **Surface basculante** : permet des animations non saccadées.

basé sur les programmes exemples des chapitres 6 et 8 où l'on trouve tous les détails sur la programmation des surfaces et du blitting.

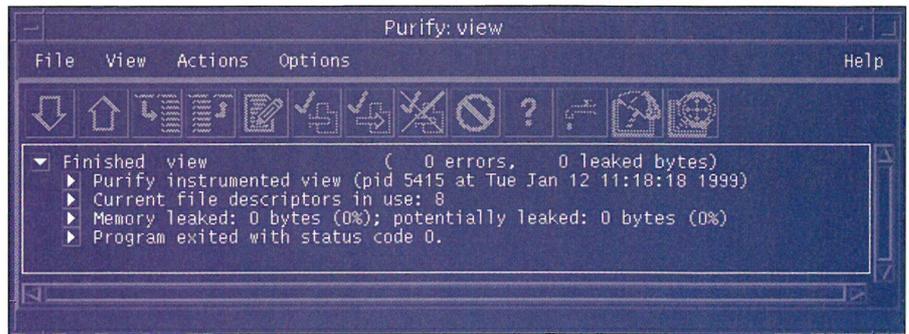
La compilation nécessite un environnement de développement comme Visual C++ de Microsoft. En plus des librairies standard, on devra y ajouter winmm.lib et ddraw.lib.

Le code présenté dans cette article ne contient pas tous les gadgets possibles que devrait présenter une application complète et robuste :

- Les différents modes graphiques auraient pu suivre un schéma similaire à celui présenté au chapitre 13 (page 20). Comme c'est le dernier chapitre consacré à DirectDraw, on peut se dire avec raison qu'on a fait le tour du sujet !
- Le clavier n'est pas accessible pendant l'exécution de la boucle et on ne peut interrompre le programme.
- Il n'y a pas de restauration des ressources au cas où l'on basculerait sous Windows (voir chapitre 12, page 195).
- Le code n'a été compilé et testé que sur une seule machine avec une carte Matrox Millennium relativement ancienne et nécessiterait certainement des corrections sur des cartes récentes G200, Riva TNT ou Banshee.

Toute l'initialisation se fait dans la fonction doInit(). Les méthodes DirectDrawCreate() et SetDisplayMode sont détaillées dans le chapitre 5.

On va créer une surface primaire et deux basculantes (voir chapitre 9) car le programme préparera l'évolution des générations. Si la machine est assez puissante et que l'on change la grandeur, le nombre ou le dessin du martien, on risque de rattraper le taux de rafraîchissement de l'écran et une deuxième (ou troisième) surface basculante nous permettra d'aller encore plus vite !



> Figure 4 : quand Purify a tout purifié, cela donne 0 erreur. Impeccable, non ?

On utilise la couleur verte comme couleur de fond : notre masque pour le dessin. Enfin, on charge notre image unique qui a été créée avec POV-RAY Tracing dans un tampon de stockage.

Lorsque le programme est lancé, il attendra une entrée de l'utilisateur : voir **tableau 1 page 60**.

## La fonction principale activée par la touche F2

La fonction F2 est la partie principale du programme. Elle montre l'évolution de la population d'une manière progressive. On voit naître ou mourir les différents individus. La fonction Vie::dimension() nous retourne une valeur progressive. Le rectangle de la surface basculante correspondant à un martien ne sera pas touché si aucun changement n'a lieu entre deux générations. Sinon, il sera effacé avec un rectangle de la grandeur de l'objet précédent (même s'il croît) et redessiné avec l'objet POV avec la dimension désirée.

Un calcul pointu (cstep) est nécessaire pour ne pas donner l'impression de dépla-

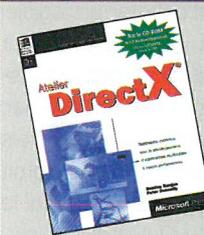
cement des objets à chaque évolution.

### Exercice 4 :

- utiliser directInput (chapitre 23 et suivants) pour y ajouter des variantes,
- en utilisant le monde fini de l'exercice 1, le faire tourner sur lui-même à une vitesse constante à l'image de la planète Mars, son home sweet home,
- temporiser par calcul (nombre d'objets et capacité de la machine) l'affichage entre chaque génération. La méthode consisterait sans doute à augmenter ou à diminuer la valeur maximum passée à Vie::dimension().

J'aurais aussi aimé trouver un battement de cœur et le synchroniser sur l'animation des générations avec directSound (chapitre 16) sur notre carte son 3D. On n'arrête pas le progrès et l'imagination qui nous aident à écrire de tels programmes ! ■

## Références bibliographiques



■ **Atelier DirectX - Microsoft Press** ISBN: 2-84082-370-5

> Nicolas Boichat, 13 ans et déjà tombé dans la marmite C++ de son Papa.

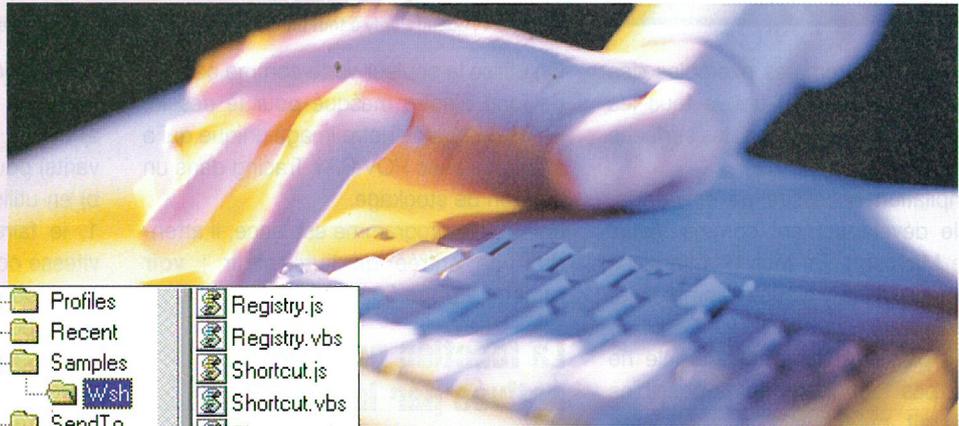


# Devenez un BATman sous Dos et Windows



Un nouveau type de programmes est en plein essor. Dérivé du Visual Basic, il risque de mettre nos bons vieux fichiers BAT hors service.

■ par Henri Ruch  
hruch@infonie.ch



Explorez votre disque dur et regardez dans le répertoire Windows s'il y a un répertoire wsh, comme Windows Host Script. Peut-être avez-vous déjà rencontré VBScript dans des pages HTML, (son utilisation la plus courante). Il correspond au VBA, avec les pages Web pour domaine d'application. Mais combien savent qu'il peut exécuter des tâches sous Windows 98 ? Inutile d'acheter un logiciel cher pour le travailler, un simple éditeur comme Notepad.exe suffit : tapez vos instructions, sauvegardez et lancez le fichier. Vous pouvez le lancer en cliquant deux fois sur l'icône. Si vous utilisez le bouton droit de la souris, vous obtiendrez *Ouvrir*, qui le lance dans *Windows* ou *Ouvrir avec l'invite MS-DOS* et *Modifier*, qui l'édite dans le Notepad. Microsoft a essayé de tout prévoir. Qu'il soit Java ou VB, un script fonctionnera dans les deux environnements. Créons notre premier programme. Ouvrez Notepad et tapez :

```
Wscript.echo "Programmez !"
```

Sauvez votre texte sous PREMIER .VBS et cliquez deux fois sur le fichier sauvé. Il s'affiche indifféremment sous Windows et sous MS-DOS. Microsoft fournit avec Windows 98 et certaines mises à jour de Windows 95 un programme capable de traiter

ces instructions. Je pense que les fichiers VBS vont vite détrôner les BAT, car ils sont simples à réaliser et surtout beaucoup plus puissants. Ils acceptent une grande partie des fonctions Visual Basic (cf. **tableau ci-dessous**). Ce petit source utilise toutes les propriétés de VBScript qui y sont citées :

### Option Explicit

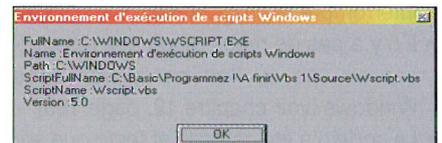
Dim S

```
S= "FullName : " + Wscript.FullName + chr(10)
S= S & "Name : " & Wscript.Name + chr(10)
```

```
S= S + "Path : " & Wscript.Path + chr(10)
S= S & "ScriptFullName : " & Wscript.ScriptFullName & chr(10)
S= S + "ScriptName : " & Wscript.ScriptName + chr(10)
S= S & "Version : " & Wscript.Version
```

Wscript.Echo S

Vous devriez obtenir sous Windows :



## 1 Wscript

## Objet d'interface entre le script et le système

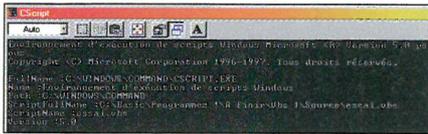
### Les méthodes-clés :

Echo texte [ &texte ][ + texte ] :	Affiche le texte dans une fenêtre Windows. L'utilisation de plusieurs paramètres est possible utilisant "&" ou "+" comme lien.
Quit [nombre]	Quitte le script actif. Si une valeur numérique est indiquée, le script donnera ce paramètre en retour.
CreateObject(<chaîne>)	Crée un objet nommé par chaîne, de type string.
DisconnectObject(<objet>)	Libère l'objet.
GetObject(<chaîne>)	Sélectionne un objet.

### Les propriétés de l'interpréteur :

FullName	Nom complet de l'interpréteur, avec son chemin.
Name	Nom de l'interpréteur.
Path	Chemin de l'interpréteur.
ScriptFullName	Nom complet du script, avec son chemin.
ScriptName	Nom du script, sans le chemin.
Version	Version de l'interpréteur.

Sous MS-DOS, vous devriez obtenir :



Notez la différence de nom : un bon moyen de savoir si le script est sous MS-DOS ou Windows. 1<sup>ère</sup> variable introduite : S. En VBScript, une variable est toujours de type Variant mais peut contenir des sous-types comme Boolean, Byte, Integer, Single, Long, Double, Date, Error, Object ou String. VBScript offre les mêmes possibilités que VB. Appelez une fenêtre de message en utilisant MsgBox avec les arguments du **tableau 2**. Bouton peut prendre les valeurs indiquées dans le **tableau 3**. Retour peut prendre les valeurs de vbOK (OK), vbCancel (Annuler), vbAbort (Abandonner), vbIgnore (Ignorer) vbRetry (Réessayer), vbYes (Oui) ou vbNo (Non). L'exemple suivant utilise MsgBox pour demander à l'utilisateur s'il faut continuer. S'il presse "non", la valeur de retour sera vbNo et le script s'arrêtera. Sinon, il affichera un message. Pour les lecteurs maîtrisant mal le Visual Basic, le \_ en fin de ligne signifie que la ligne d'instruction n'est pas terminée.

```
A = B _
+ C _ Equivaut à A=B+C
```

Cela facilite la lecture d'un source.

### Option Explicit

```
Dim IntRep
IntRep = MsgBox("Continuer ?", _
vbYesNo _
+ vbQuestion, _
"Programmez !")
If IntRep = vbNo Then
WScript.Quit
else
WScript.Echo "Bonjour"
End If
```

On peut donc aisément dialoguer avec l'utilisateur. Il est aussi possible d'attendre une réponse de sa part. Pour ce faire, employons InputBox : InputBox(Texte[, Titre] [, Default]) Retour, où Texte est la chaîne affichée en message dans la boîte de dialogue, Titre celle de sa barre de titre, Default

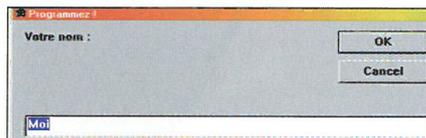
## 2 Comprendre Function MsgBox(Texte[, bouton] [, titre]) Retour

<b>Texte</b>	Chaîne affichée comme message dans la boîte de dialogue
<b>Bouton</b>	Valeurs indiquant le type de boutons à afficher, le style d'icone à utiliser, le bouton par défaut et la modalité du message
<b>Titre</b>	Chaîne affichée dans la barre de titre de la boîte de dialogue.
<b>Retour</b>	Valeur indiquant quel bouton a été pressé

## 3 Bouton Toutes les variétés disponibles

vbOKOnly	Bouton OK
vbOKCancel	Boutons OK et Annuler
vbAbortRetryIgnore	Bouton Abandonner, Réessayer et Ignorer
vbYesNoCancel	Boutons Oui, Non et Annuler
vbYesNo	Boutons Oui et Non
vbRetryCancel	Boutons Réessayer et Annuler
vbCritical	L'icone Message critique
vbQuestion	L'icone Requête d'avertissement
vbExclamation	L'icone Message d'avertissement
vbInformation	L'icone Message d'information
vbDefaultButton1	Le premier bouton est le bouton par défaut
vbDefaultButton2	Le deuxième bouton est le bouton par défaut
vbDefaultButton3	Le troisième bouton est le bouton par défaut
vbDefaultButton4	Le quatrième bouton est le bouton par défaut
vbApplicationModal	L'utilisateur doit répondre au message affiché dans la zone de message avant de pouvoir continuer de travailler dans l'application en cours
vbSystemModal	Toutes les applications sont interrompues jusqu'à ce que l'utilisateur réponde au message affiché dans la zone de message

celle affichée par défaut dans la zone de texte et Retour celle qui y affichée si l'utilisateur presse OK.



### Option Explicit

```
Dim S
S = InputBox("Votre nom :", "Programmez !", "Moi")
If S = "" Then
WScript.Quit
else
WScript.Echo("Bonjour " + S)
end if
```

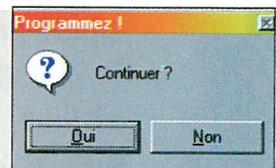
Pour les lecteurs qui ne connaissent pas le Basic, une fonction est définie par Sub et une procédure par Function. Leur emploi est très aisé. Ils peuvent être situés n'importe où dans le fichier.

### Option Explicit

```
Sub Affiche(Texte)
WScript.Echo(Texte)
End Sub
Dim S
Affiche("Bonjour")
S = Demande("Ton Nom")
```

Affiche(S)

```
Function Demande(Texte)
Demande = InputBox("Votre
nom : " _
, "Programmez !" _
, "Moi")
End Function
```



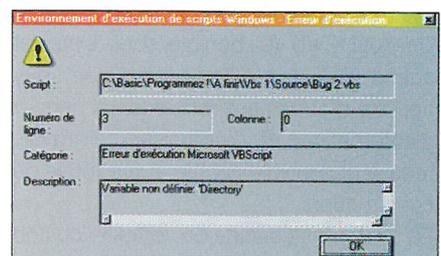
La 1<sup>ère</sup> ligne exige que toutes les variables soient déclarées avant d'être utilisées : ce n'est pas grave dans de petits scripts mais obligatoire pour les grands. Cela peut éviter des erreurs de programmations dues à des fautes de frappes. Par exemple :

```
Directory= Wscript. Path
Wscript.echo(Directory)
```

Si vous le lancez, une fenêtre vide va s'afficher. Essayez de mettre en tête du script :

```
Option Explicit
```

Et relancez-le. Une erreur d'exécution va s'afficher, vous évitant une erreur indétectable qui peut changer tout le comportement de votre script : selon sa taille, il vous faudrait des heures pour la trouver.



NB : les ( ) ne sont pas vitales. Wscript.echo(Directory) et Wscript.echo Directory réagiront strictement de même. ■

# Mon Make à moi

Programmez, programmez, il en restera toujours quelque chose, par exemple, un ensemble de fichiers sources, correspondant à autant de modules dans votre application. Un utilitaire star des GNU, nommé Make, va relier tous ces listings, en tenant compte de vos préférences pour une compilation optimale. Explorons ses principales fonctionnalités.



■ par Jérôme Arnaud  
jarnaud@usa.net

**T**out programmeur le sait, dans le monde du développement, il faut diviser pour mieux régner. Dans les projets volumineux, il n'est pas raisonnable d'écrire un seul programme contenant tout le source. Ceci dit, la programmation modulaire amène une lourdeur, liée aux tâches de compilation.

Un outil GNU (disponible sous Linux et autres plateformes Unix, ainsi que sous nombre d'autres environnements) permet d'automatiser et d'optimiser ce travail, réduisant sensiblement la durée totale de construction.

En effet, Make décide de compiler ou non chaque unité de compilation, en établissant les liens de dépendance avec les autres unités de compilation et en comparant la date du fichier objet correspon-



## C'est l'histoire d'un Make...

- Par défaut, Make utilise le fichier correspondant au nom de "Makefile". L'option -f permet d'en spécifier un autre.
- Il est possible de construire une autre cible que la première du fichier makefile : il suffit de la spécifier. Par exemple : make menu.
- L'option -s inhibe l'affichage des commandes avant exécution.
- Lorsque Make rencontre une erreur de compilation, il abandonne son travail. L'option -k lui indique de poursuivre la construction des autres cibles indépendantes du fichier source erroné.
- L'option -n permet d'afficher les commandes destinées à générer une cible sans les exécuter (pratique pour déboguer un makefile).
- L'option -d fait de Make un outil plus bavard.
- Finalement l'option -j (disponible sur certains systèmes uniquement) permet de réaliser une compilation multitâche (plusieurs compilations simultanées). Son efficacité est variable d'un système à l'autre.

dant (s'il existe) avec la date de dernière modification des sources dont il dépend. Prenons un exemple simple : un jeu d'échecs.

Sa compilation peut s'apparenter à l'ensemble de commandes suivantes :

```
gcc -c interface_graphique.c
gcc -c module_sonore.c
gcc -c min_max.c
gcc -c moteur_de_jeu.c
gcc -c main.c
gcc -o echecs main.o min_max.o module_sonore.o
interface_graphique.o -lm
```

Les dépendances sont limpides. A dépend de B si une modification de B affecte A. En langage programmeur, *interface\_graphique* dépend de *module\_sonore* ou *moteur\_de\_jeu* dépend de *min\_max* ou encore *main* dépend de tous les autres modules.

Ainsi, si on modifie *interface\_sonore.h*, on doit recompiler *interface\_sonore.c* bien sûr, mais également *interface\_graphique.c* et refaire l'édition de liens.

D'un point de vue plus théorique, l'analyse qui doit être réalisée lors d'une compilation est la suivante : élaborer un graphe de dépendance entre les fichiers et le parcourir pour établir quels sont les fichiers à compiler ou à recompiler.

Un premier makefile se présentera de la manière suivante :

```
echecs: main.o min_max.o module_sonore.o interface_graphique.o
gcc -o echecs main.o min_max.o module_sonore.o
```

## Vocabulaire de Make

- Un COMMENTAIRE est une ligne commençant par le caractère '#', ignoré par Make. Il permet de documenter vos fichiers makefile.
- Une REGLE EXPLICITE décrit comment construire un ou plusieurs fichiers, appelés CIBLES de ladite règle. Dans l'exemple ci-dessus, cinq règles explicites sont présentes. Une règle explicite est composée de deux parties : une première décrivant les dépendances et une seconde fournissant la commande à exécuter pour produire la cible.
- Une REGLE IMPLICITE décrit de manière générique comment construire des cibles, sur la base de leurs noms. Ces règles permettent d'éviter l'écriture de nombreuses règles explicites semblables et entraînent ainsi un gain de temps pour le concepteur de makefile.
- Une DEFINITION DE VARIABLE associe une chaîne de caractères à un nom de variable. Au moment de l'exécution des commandes de compilation, chaque variable est remplacée par sa valeur. Elles sont donc très utiles pour positionner par exemple les options de compilation.

```
xterm
[jarnaud@helium linux]$ make clean
make[1]: Entering directory `/usr/src/linux-2.0.36/arch/i386/boot'
rm -f bootsect setup
rm -f bbootsect
rm -f zImage tools/build compressed/vmlinux.out
rm -f bzImage tools/bbuild compressed/bvmlinux.out
make[2]: Entering directory `/usr/src/linux-2.0.36/arch/i386/boot/compressed'
rm -f xtract piggyback vmlinux bvmlinux
make[2]: Leaving directory `/usr/src/linux-2.0.36/arch/i386/boot/compressed'
make[1]: Leaving directory `/usr/src/linux-2.0.36/arch/i386/boot'
make -C arch/i386/kernel clean
make[1]: Entering directory `/usr/src/linux-2.0.36/arch/i386/kernel'
rm -f trampoline trampoline.hex hexify
make[1]: Leaving directory `/usr/src/linux-2.0.36/arch/i386/kernel'
rm -f kernel/ksyms.lst include/linux/compile.h
rm -f core `find . -name '*.oas' ! -regex '.*\xdialog/.*' -print`
rm -f core `find . -type f -name 'core' -print`
rm -f vmlinux System.map
rm -f .tmp* drivers/sound/configure
rm -fr modules/*
rm -f submenu*
[jarnaud@helium linux]$
```

> Un Make "clean" fera tout le travail de compilation que vous avez préalablement configuré.

```
interface_graphique.o -lm
main.o: main.c interface_graphique.h
moteur_de_jeu.h

gcc -c main.c

interface_graphique.o: interface_graphique.c interface_graphique.h module_sonore.h

gcc -c interface_graphique.c

module_sonore.o: module_sonore.c module_sonore.h

gcc -c module_sonore.c

moteur_de_jeu.o: moteur_de_jeu.h moteur_de_jeu.c
min_max.h

gcc -c moteur_de_jeu.c

min_max.o: min_max.h min_max.c

gcc -c min_max.c
```

Il convient d'écrire ce fichier "Makefile" dans le répertoire contenant les sources et de lancer "make".

Lors de la première compilation, tous les objets sont créés et l'édition de lien donne naissance à un exécutable.

Si vous exécutez "make" à nouveau, aucune action n'est réalisée car il n'y a pas eu de modifications.

Si, par contre, on modifie *min\_max.h*, Make va constater que sa date de modification est plus récente que celle de *main.o*. Il va donc réaliser la commande suivante :

```
gcc -c min_max.c
```

mais également :

```
gcc -c moteur_de_jeu.c
```

ainsi que :

```
gcc -o echecs main.o min_max.o module_sonore.o
interface_graphique.o -lm
```

pour produire un nouvel exécutable.

## Un Make en règles

Comme nous venons de le voir dans l'exemple précédent, les règles, qu'elle soient explicites ou implicites, reprennent toujours la même forme sémantique, indiquée page suivante.

```

Buffers Files Tools Edit Search Mule Makefile Help
*****
# Generic JAVA Makefile #
*****

JAVAC      = /usr/local/java/bin/javac
JAVA_SRCS  = $(notdir $(wildcard *.java))
JAVA_OBJS  = $(patsubst %,java,%.class,$(JAVA_SRCS))

all        : $(JAVA_OBJS)
%.class    : %.java
            $(JAVAC) $<

clean      :
            rm -f *.class

```

> Le makefile marche également en Java, comme avec les autres langages référençables par le projet GNU.

#### CIBLES : LISTE DE DÉPENDANCES

#### COMMANDE

...

D'un point de vue syntaxique, cibles et dépendances doivent se trouver sur la même ligne (éventuellement, un caractère '\ ' permet d'ignorer la coupure). Les différentes dépendances de la liste sont séparées par un espace. Chose importante (source de nombreuses erreurs lorsque l'on débute), la ligne de commande doit obligatoirement commencer par une tabulation, sans quoi, un 'missing operator' vous rappelle à l'ordre. Plusieurs commandes peuvent se succéder.

D'un point de vue sémantique, un fichier ne peut apparaître qu'une seule fois comme cible dans les règles d'un makefile. Avant d'exécuter une règle, Make parcourt la liste des dépendances. Si une d'entre elles est une cible d'une autre règle, make applique d'abord cette nouvelle règle.

Ce processus étant récursif, Make exécute donc en premier les commandes des règles dont les dépendances n'apparaissent dans aucune cible.

## Les commandes

Le comportement par défaut de Make est d'afficher les commandes avant de les

exécuter. Ceci dit, il est possible de le modifier par le biais d'un caractère de contrôle inséré dans l'en-tête d'une commande.

Par exemple :

```

salutSylvia : coucou.o sylvia.o

@echo 'Edition de liens'

gcc -o salutSylvia coucou.o sylvia.o

```

Le caractère '-' peut également être placé en en-tête d'une commande. Dans ce

cas, si celle-ci échoue, Make n'interrompt pas son travail (son comportement par défaut) pour autant et passe à la commande suivante.

Finalement, chaque ligne de commande est exécutée par un interpréteur de commande. Un makefile comme suit peut donc produire une erreur :

```

direBonjour/salutSylvia : coucou.o sylvia.o

cd direBonjour

gcc -o salutSylvia ../coucou.o ../sylvia.o

```

Il faudrait en effet que ce soit le même interpréteur qui exécute les deux commandes. Pour cela, il est possible d'insérer à la fin de la première commande un point-virgule (séparateur de commande), suivi du caractère '\ ' qui permet de poursuivre une commande longue sur la ligne suivante.

## Les variables

Les variables sont plus proches de ce que l'on nomme macro-définitions dans certains langages, comme le C ou le Lisp, que les véritables variables de ceux-ci. Elles permettent de représenter un nom d'exécutable, une directive de compilation, un ensemble de répertoires ou toute autre chose encore.

Une variable est donc un identificateur au sens de la majorité des langages (pas de ':', ',', '#', '=', espace...), elle est traditionnellement composée de caractères en

## Make it easy

Le respect de certaines conventions facilite l'écriture ainsi que la lecture des makefile.

#### ■ Les cibles standard :

all: -> toujours placé en première règle explicite et réalisé par défaut.

install: -> installation (généralement copie) du programme produit par Make.

uninstall: -> annule les actions de install.

clean: -> supprime tous les fichiers produits par Make.

#### ■ Usage des variables :

Utiliser au maximum les variables pour permettre un paramétrage plus facile. Généralement, on donne aux variables le nom de l'utilitaire standard en majuscule. Exemples : \$(CC), \$(CP)...

Pour faciliter la réutilisation du makefile, il convient également d'utiliser les variables en lieu et place des options, paramètres et répertoires.

majuscules. C'est une convention. Pour définir la valeur d'une variable, il suffit de respecter la forme simple : VARIABLE = VALEUR. Quelques exemples :

```
OPT = -Wall
OPTIONSDEBUG = -DDEBUG -g
DEPS = moteur.o carrosserie.o roues.o
```

Il est possible d'ajouter un élément à une variable :

```
DEPS += autoradio.o
```

La valeur d'une variable est fournie par \$(VARIABLE).

Exemple :

```
gcc -c $(CFLAGS) prog.c
```

Il existe également un ensemble de variables automatiques gérées directement par Make :

\$@ : nom de cible de la règle.  
 \$< : nom de la première dépendance.  
 \$? : nom de toutes les dépendances plus récentes que la cible.  
 \$^ : nom de toutes les dépendances d'une cible.  
 \$\* : radical d'une règle implicite (pour un motif %.o et une cible prog.o, \$\*, le radical a pour valeur toto).  
 Ces différentes variables permettent l'écriture des règles implicites.

## Les règles implicites

La structure d'une règle implicite est la même que celle d'une règle explicite. Ceci dit, un caractère motif '%' s'associe à n'importe quelle chaîne non vide. Lorsqu'une chaîne a été sélectionnée par un motif, '%' est remplacé par le radical dans les dépendances de cette règle. Ainsi, par exemple :

```
%.o : %.c
$(CC) -c -o $@ $(CFLAGS) $(CPPFLAGS) $<
```

permet de produire n'importe quel objet à partir de son source. Voyons l'écriture des liens génériques :

```
%.o $(LIBS)
$(CC) -o $@ $(LDLAGS) $< $(LIBS)
```

## Les directives

Les diverses versions de Make proposent différentes directives. Une seule est commune (norme POSIX) : la directive d'inclusion. Ecrire "Include fichier" permet d'inclure un fichier dans un makefile et autorise ainsi leur écriture modulaire. ■



> Le logo des logiciels libres GNU.

**Dossier spécial : TRICHER POUR GAGNER !**

# PC LOISIRS

mag

Avril 1999  
N°8

## SPECIAL SOLUTIONS

**60 PAGES de soluces**

CONSEILS ET TIPS POUR :

- Tomb Raider 3 p.12
- BlackStone Chronicles p.60
- SIM CITY 3000 p.42
- Alpha Centauri p.66
- Dark Project p.50
- Settlers 3 p.56
- ODT p.28

BarCode 150 82 - Suisse 4 5075 - Canada 1 853 3 CAN  
M 8548 - 8 - 29,00 F - RD

# Bloqué ? La solution existe !

Avec PC mag LOISIRS, vous bénéficiez de dizaines de pages de soluces, d'aides, d'astuces de jeu et des meilleurs conseils de la rédaction pour jouer jusqu'au bout !

**EN VENTE ACTUELLEMENT CHEZ VOTRE MARCHAND DE JOURNAUX**

# Imprimez les répertoires du disque dur

Avoir une trace papier d'un CD, d'une disquette ou d'une partie de votre disque dur, vous ferait gagner beaucoup de temps. Cette fonctionnalité, absente dans Windows, est comblée par ce logiciel en VB.

■ par Henri Ruch  
hruch@infonie.ch



Il vous est certainement déjà arrivé de chercher un programme précis sur un CD-ROM. La dernière fois que vous l'avez utilisé, vous étiez sûr de le retrouver facilement en cas de besoin. Mais voilà, sur quel disque était-il ?

Il ne vous reste plus qu'à rechercher lequel de vos 118 CD-ROM contient ce que vous cherchez. Et si vous pouviez imprimer une partie du contenu de ces disques ?

L'utilisateur doit pouvoir se promener librement sur ces divers lecteurs, disques durs et lecteurs de CD-ROM. Après avoir choisi son répertoire, il doit aussi pouvoir imprimer son contenu. Il serait bon de pouvoir lui laisser la possibilité de définir s'il veut développer les sous-dossiers

et s'il désire afficher les fichiers qui s'y trouvent.

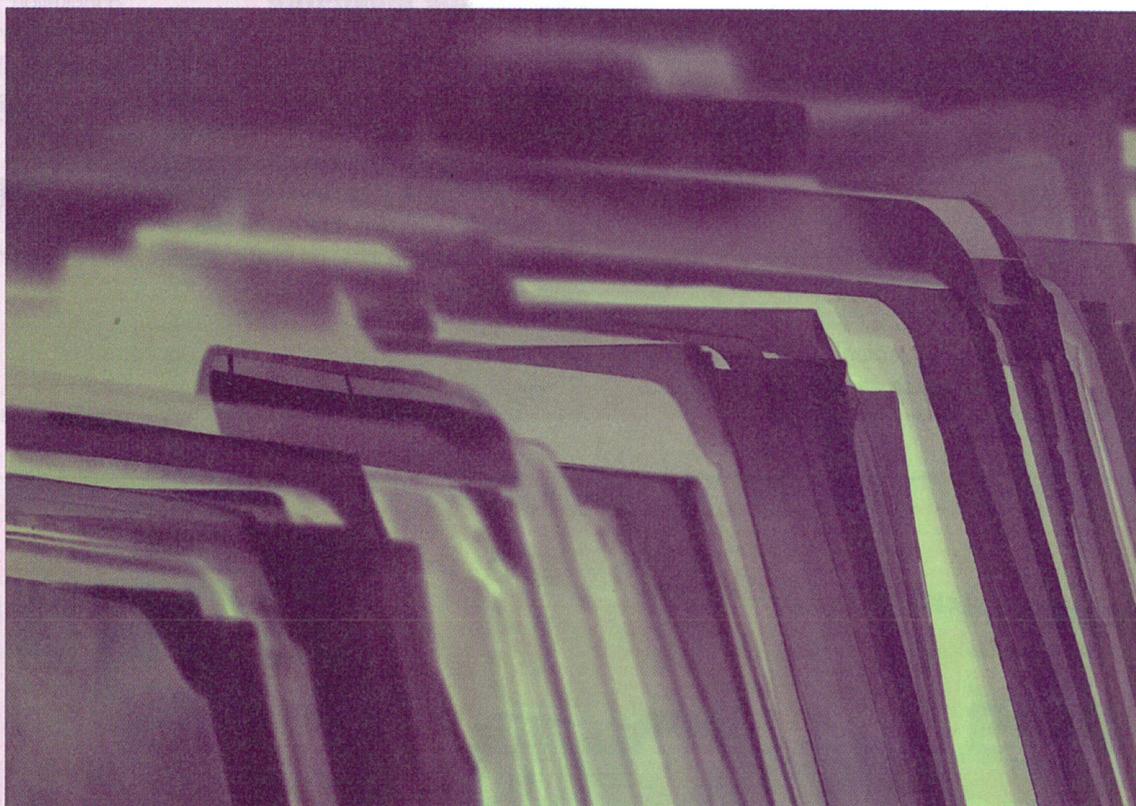
Laissons également à l'utilisateur le choix de définir s'il désire y voir apparaître la date et la taille des fichiers. Offrons-lui encore la possibilité de modifier la taille, le type et la couleur de la fonte qui sera imprimée. Qu'il puisse observer à l'écran ce qui va être imprimé serait un plus non négligeable.

## Une paisible promenade sur le disque dur

Pour répondre à tous ces critères, donnons-lui cet aspect : la partie de gauche permettra une promenade complète sur les disques de l'utilisateur. Il pourra ainsi

déterminer le répertoire à imprimer. Plus bas, des cases à cocher nous permettront de connaître ses désirs. A droite, la page qui devra être imprimée est montrée à l'écran. L'encadré ci-contre décrit les fonctionnalités des boutons.

Vous constaterez que deux options ont été ajoutées à l'utilitaire. La page est toujours tracée sur un objet de dessin pour se figurer le résultat de l'impression. Il suffit de deux à trois lignes de commande pour sauver cette image sur le disque, alors pourquoi s'en priver. Il en va de même avec la fonction permettant de sauvegarder le contenu du répertoire dans un fichier de texte. Toutes les informations nécessaires sont stockées en mémoire vive.



## Navigation à vue

Permettre à l'utilisateur de naviguer entre les différents lecteurs ou sur les différents disques durs peut se résumer à deux objets :

- un DriveListBox pour choisir entre les divers lecteurs existants sur le disque dur de votre ordinateur,
- un DirListBox pour naviguer sur le lecteur sélectionné.

La liaison entre ces deux objets se fait tout simplement dans Lecteurs\_Change, où la propriété Drive du DriveListBox est attribuée à la propriété Path du DirListBox.

## Les options en série

Comme vous pouvez le constater sur l'image de la fenêtre principale, quatre options sont proposées :

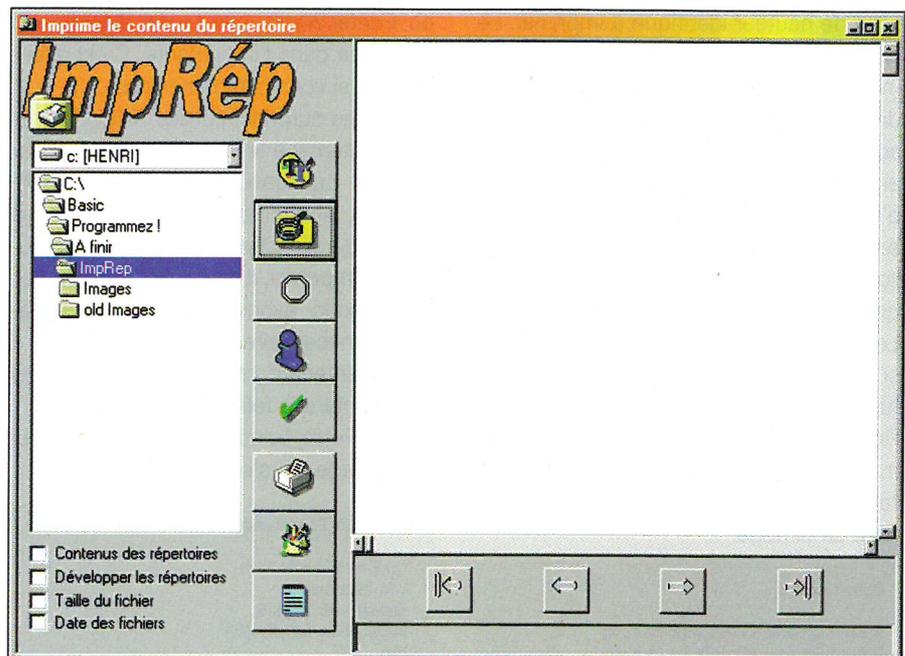
- Contenu du répertoire.
- Développer les répertoires.
- Taille des fichiers.
- Date des fichiers.

L'activation ou la désactivation des deux premières options efface toutes les données lues précédemment. C'est logique car lors de la recherche sur le disque, la routine tient compte de ces deux options et ne lit que les renseignements exigés.

La modification des deux dernières options ne provoque qu'une nouvelle écriture de la page actuelle. Ces options

### Les boutons et leurs fonctionnalités

	Couleur, police et taille.
	Lancer la recherche.
	Arrêter la recherche.
	Information sur l'utilitaire.
	Quitter l'utilitaire.
	Imprimer la page.
	Sauvegarder la page en bmp.
	Sauver les données en texte.



> Naviguez facilement dans les dossiers du disque dur et imprimez-les.

sont sauvegardées sur le disque lors de l'abandon du programme. Regardez simplement Form\_Unload et Enregistrer situés dans FrmImpRep. La récupération se fait lors de la mise en route du logiciel, dans Form\_Load et Lire.

## Une bonne impression

Il existe peu d'utilitaires faits par des programmeurs non professionnels qui offrent une possibilité d'impression de bonne qualité. Souvent, cette partie est réduite à l'impression de quelques lignes, sans mise en page et sans esthétique. Pourtant, cela n'est pas aussi compliqué qu'il n'y paraît. Cela vient peut-être du fait que le manuel de l'utilisateur n'aborde que très brièvement, et à mon avis très mal, ce sujet.

Mais en fait en quoi consiste une impression ? La manière la plus aisée de l'aborder est de comparer la page que l'on veut imprimer à une image vierge, avec une taille définie par les obligations physiques de votre feuille de papier. Vous pouvez dessiner, écrire du texte ou tirer un trait sur cette page comme vous le feriez sur une bitmap normale.

La seule différence qui existe, c'est que le résultat ne sera visible sur le papier qu'une fois que la totalité des opérations sera terminée et que l'ordre aura été

donné à l'imprimante de l'écrire. Examinez attentivement l'objet Printer dans l'aide de VB5, avant de poursuivre la lecture de cet article. Je vais boire un petit café en attendant.

Vous avez remarqué qu'il est possible d'obtenir de nombreux renseignements sur la feuille de papier sélectionnée. Avec Height et Width, on obtient facilement ces indications. Si vous avez observé les différentes méthodes, vous avez pu constater qu'il existe Line, Circle... exactement comme sur des objets PictureBox. Il possède cependant trois méthodes de plus que ces derniers, à savoir NewPage, EndDoc, Killdoc. Comme leur nom l'indique, ils servent à créer une nouvelle page, à terminer un document et à supprimer un document.

## Aperçu en bitmap

Le principe de base d'un aperçu avant impression est de dessiner sur un bitmap ce qui va être imprimé sur la feuille de papier. En principe, les mêmes règles de mise en page vont s'appliquer aux deux éléments. Une page se compose en principe d'un en-tête, d'un pied de page et d'une partie centrale.

Comme le nombre d'éléments à imprimer est connu, le calcul du nombre de pages nécessaires est aisé ! Dans notre cas,

chaque dossier ou fichier représente une ligne de hauteur identique à imprimer. Le nombre de lignes qu'il est possible de mettre sur la page sera le résultat de la division de la hauteur de page, dont on aura bien sûr enlevé la place nécessaire au bas et au haut de page, par la hauteur d'une ligne. Le nombre de pages nécessaires s'obtient en divisant le nombre de lignes à afficher par le nombre de lignes qu'il est possible de placer sur une page. Le calcul s'avérerait bien plus complexe si les lignes n'avaient pas toutes la même hauteur.

Regardons un peu comment le logiciel traite ce problème. Dans un premier temps, il récupère les informations sur la taille de la page.

## Option Explicit

```
Public LngHauteurPageTwips As Long
Public LngLargeurPageTwips As Long
```

```
Private Declare Function GetDeviceCaps Lib "gdi32"
(ByVal hdc As Long, ByVal nIndex As Long) As Long
Private Pixel As Integer
Const HORZRES = 8
Const VERTRES = 10
```

```
Public Sub Info_imprimante()
```

```
    ' Renseignements sur la largeur et hauteur de la page
```

```
    Pixel = Printer.ScaleX(1, vbPixels, vbTwips)
    Printer.KillDoc
    LngLargeurPageTwips =
    EnTwips(GetDeviceCaps(Printer.hdc, HORZRES))
    LngHauteurPageTwips =
    EnTwips(GetDeviceCaps(Printer.hdc, VERTRES))
End Sub
```

```
Public Function EnTwips(IntPixel As Variant) As
Integer
    EnTwips = IntPixel * Pixel
End Function
```

La largeur et la hauteur de la page sont récupérées grâce à la fonction de WinApi32 GetDeviceCaps. Cette dernière renvoie ces mesures en pixels. EnTwips ne fait que transformer cette valeur en TWIPS. Oui, je sais, vous êtes surpris par deux choses. Il y a quelques lignes, je vous parlais de l'objet Printer qui nous renvoyait, avec Width et Height le format de la page et maintenant, j'utilise GetDeviceCaps dont les fonctions sont indiquées plus bas (voir tableau 1).

Il existe une trentaine de constantes permettant de récupérer des informations différentes. Je ne vous en cite que quelques-unes qui pourront vous servir (voir tableau 2).

Ce choix s'est imposé de lui-même. Si vous essayez d'avoir les renseignements

sur la taille de la page par Printer.Height ou Printer.Width, une feuille blanche est introduite dans l'imprimante. C'est assez gênant, si vous n'imprimez pas.

Deuxième surprise, j'utilise KillDoc après avoir déterminé le nombre de TWIPS par PIXELS. Là aussi, l'objet Printer a une réaction surprenante. Il n'imprime pas une page blanche, mais crée un document à imprimer. Il faut donc le supprimer. Voilà, nous avons tous les renseignements nécessaires sur l'imprimante. Il faut maintenant créer la page écran qui représentera la feuille de papier.

C'est Form\_Load de FrmImpRep et EffacePage qui s'en chargent :

```
Private Sub Form_Load()
```

```
    On Error Resume Next
    Info_imprimante
    Espace = EnTwips(5)
    IntGauche = EnTwips(20)
    Lire
    OperationEnCours = False
    Page.AutoSize = True
    InitialisePage
    CmdStop.Enabled = False
    AnalyseOptions
```

```
    PctBarre.AutoRedraw = True
    PctBarre.ScaleWidth = 100
    PctBarre.DrawMode = vbNotXorPen
```

```
    DessinePage Page, PageEnCours
End Sub
```

- La fonction prend les renseignements sur l'imprimante et la feuille de papier.
- Certaines marges sont définies.
- Il faut encore lire les options définies lors de la dernière utilisation. Nous en avons déjà parlé plus haut.
- Page, qui sera l'objet qui représentera la feuille de papier, est mis en mode AutoSize pour pouvoir le redimensionner.
- Il est initialisé.
- La barre de défilement est initialisée. Je ne parlerai pas de cette barre car elle a été expliquée en détail dans le dernier article.
- Enfin la page est dessinée.

```
Private Sub EffacePage()
```

```
    Page.Picture = LoadPicture()
    Page.Cls
    Page.DrawWidth = 1
    Page.Width = LngLargeurPageTwips
    Page.Height = LngHauteurPageTwips
    Page.ForeColor = LngCouleur_fonte
    Adapte
End Sub
```

- L'objet Picture de Page est effacé.
- L'objet Image de Page est effacé. Il faut savoir qu'il y a une différence entre ces deux objets. Si vous dessinez sur Page, vous influencez en fait Image. Si une image est chargée dans Picture, Image en sera de même influencé. Faites de petits essais, vous verrez, c'est parfois très drôle.
- La largeur du trait est ensuite définie.
- La grandeur de Page est adaptée à celle de la feuille de papier.

A partir de maintenant, vous avez la possibilité de reproduire exactement ce qui va être imprimé sur votre objet Page. Votre Aperçu avant Impression vient d'être créé. Son affichage ne pose aucun problème. N'aimant pas inventer ce qui l'est déjà, j'ai simplement utilisé la technique décrite dans le manuel de l'utilisateur page 309 de l'édition française.

## Une mise en page attirante

Je résume, en quelques lignes, le chemin que nous avons suivi :

- Tout d'abord, la prise de renseignements sur la feuille de papier avec GetDeviceCaps.
- Le dimensionnement d'un objet PictureBox selon les renseignements obtenus.
- Enfin, l'affichage de cet objet avec la technique enseignée dans le guide de l'utilisateur.

La page ainsi créée, il faut encore calculer ce que l'on peut y mettre et le nombre de pages nécessaires pour imprimer tous

Tableau 1 : la fonction GetDeviceCaps

Code	Signification
Function GetDeviceCaps (	Déclaration
ByVal hdc As Long,	Dc du périphérique dont on cherche les informations
ByVal nIndex As Long	Type d'informations à récupérer
)As Long	Selon les informations demandées

**Tableau 2 : les constantes du programme**

HORZSIZE = 4	Largeur d'affichage en millimètres
VERTSIZE = 6	Hauteur d'affichage en millimètres
HORZRES = 8	Largeur d'affichage en pixels
VERTRES = 1	Hauteur d'affichage en pixels

les renseignements demandés. J'aime ce qui est beau, non seulement dans la nature mais aussi en informatique. Une belle mise en page sera plus attirante qu'une suite de lignes sur une feuille blanche. Nous allons donc créer un en-tête et un pied de page. Ceci fait, il ne nous restera plus qu'à déterminer le nombre de lignes que l'on peut mettre sur la place encore libre et calculer le nombre de pages nécessaires pour tout imprimer.

La hauteur nécessaire à l'affichage de cet en-tête sera calculé par CalculeEntete :

```
Private Function CalculeEntete(But As Object) As Integer
' Calcule la place nécessaire à l'affichage de l'en-tête
Dim IntHaut1 As Integer
Dim IntHaut2 As Integer

IntHaut1 = (But.TextHeight("WW") * 2) + (2 * Espace)
IntHaut2 = Espace + PtLogo.Height + (3 * Espace)
CalculeEntete = IntHaut2 + Espace
If IntHaut1 > IntHaut2 Then
CalculeEntete = IntHaut1 + Espace
End If
End Function
```

- La hauteur nécessaire à l'affichage du texte est d'abord calculée. TextHeight nous fournit simplement la hauteur de la chaîne de caractères qui lui est transmise.
- Vient ensuite le calcul de la hauteur de l'image.
- C'est la plus grande des deux valeurs qui est retenue comme hauteur effective du haut de la page.

Le bas de page sera moins complexe. Le calcul de la place nécessaire n'en sera que simplifié.

```
Private Function CalculeBas(But As Object) As Integer
' Calcule la hauteur nécessaire
' Pour le bas de la page
Dim IntTaille As Integer
IntTaille = But.TextHeight("Page 9/9")
IntTaille = IntTaille + (2 * Espace)
CalculeBas = IntTaille
End Function
```

Il suffit de calculer la hauteur du texte en fonction de la fonte employée, en n'ou-

bliant pas les espaces nécessaires en dessus et en dessous de "page".

Les différentes routines de calcul de place étant posées, le reste des informations va pouvoir être trouvé rapidement.

```
Private Sub calcule_page()
' Calcule le nombre de lignes par page

' Calcule le nombre de pages nécessaires
' pour afficher toutes les données

Page.FontName = StrNomFonte
Page.FontSize = IntTaille_fonte
Printer.FontName = StrNomFonte
Printer.FontSize = IntTaille_fonte

IntHauteurLigne = Page.TextHeight("WW") +
EnTwips(10)
LigneParPage = (Page.Height - CalculeEntete(Page) -
CalculeBas(Page)) \ IntHauteurLigne
IntNombrePage = Nombre_Chemin \ LigneParPage
If IntNombrePage * LigneParPage <>
Nombre_Chemin Then IntNombrePage =
IntNombrePage + 1
VScroll1.LargeChange = IntHauteurLigne * 2
VScroll1.SmallChange = IntHauteurLigne
End Sub
```

- La fonte choisie est attribuée à la page.
- La hauteur d'une ligne est calculée en fonction de cette dernière, avec un

espacement de cinq pixels en dessus et en dessous.

- Le nombre de lignes par page est le résultat de la place disponible sur la page, soit sa hauteur moins l'en-tête et le bas de page, divisé par la hauteur d'une ligne.
- Le nombre de pages nécessaires se trouve en divisant le nombre total de lignes à afficher par le nombre de lignes que l'on peut mettre sur une feuille.
- Travaillant avec des nombres entiers, il faut encore contrôler que la fraction disparue dans la division est mémorisée dans le nombre de pages. Car si le résultat de la division est de 3,25 le nombre qui en découlera sera 3.

## Dessine-moi une page

La page sera simplement dessinée avec la routine DessinePage :

```
Private Sub DessinePage(But As Object, num As Integer)
' Dessine la page
Dim IntCompteur As Integer
Dim LngHaut As Long
Dim IntDepart As Integer
Dim IntNombre As Integer

HScroll1.Value = 0
VScroll1.Value = 0
PageEnCours = num
IntDepart = ((num - 1) * LigneParPage) + 1
IntNombre = num * LigneParPage
If IntNombre > Nombre_Chemin Then IntNombre =
```



> Le module d'impression simplissime mais efficace pour éviter le gaspillage de papier.

```

Nombre_Chemin
Initialise_Barre (IntNombre - IntDepart)
LngHaut = AfficheEntete(But)
For IntCompteur = IntDepart To IntNombre
DoEvents
Progresse (IntCompteur - IntDepart)
DessineLigne But, IntCompteur, LngHaut
LngHaut = LngHaut + IntHauteurLigne
If BolArrete = True Then
Initialise
Exit Sub
End If
Next IntCompteur
PctBarre.Cls
AfficheBas But
End Sub

```

Qui fait appel à AfficheEntete :

```

Private Function AfficheEntete(But As Object) As Integer
'Dessine le haut de la page

Dim IntTaille As Integer
Dim IntPlaceReste As Integer
Dim IntGauche_1 As Integer
Dim IntLarge As Integer

IntTaille = CalculeEntete(But) - Espace
But.CurrentX = IntGauche
But.CurrentY = Espace
But.PaintPicture PtLogo.Picture, But.CurrentX,
But.CurrentY
But.Line (0, IntTaille)-(But.Width, IntTaille)
AfficheEntete = IntTaille + Espace

IntPlaceReste = But.Width - PtLogo.Width - (2 *
Espace)
IntGauche_1 = IntGauche + PtLogo.Width + (2 *
Espace)

```

```

But.FontBold = True
IntLarge = But.TextWidth("Contenu du répertoire :")
If IntLarge < IntPlaceReste Then
IntLarge = (IntPlaceReste - IntLarge) \ 2
But.CurrentX = IntGauche_1 + IntLarge
But.CurrentY = Espace
But.Print ("Contenu du répertoire :")
End If
But.FontBold = False
IntLarge = But.TextWidth(Chemins.Path)
If IntLarge < IntPlaceReste Then
IntLarge = (IntPlaceReste - IntLarge) \ 2
But.CurrentX = IntGauche_1 + IntLarge
But.CurrentY = Espace + But.TextHeight("WW")
But.Print (Chemins.Path)
End If
End Function

```

Et à AfficheBas :

```

Private Sub AfficheBas(But As Object)
'Affiche le bas de la page

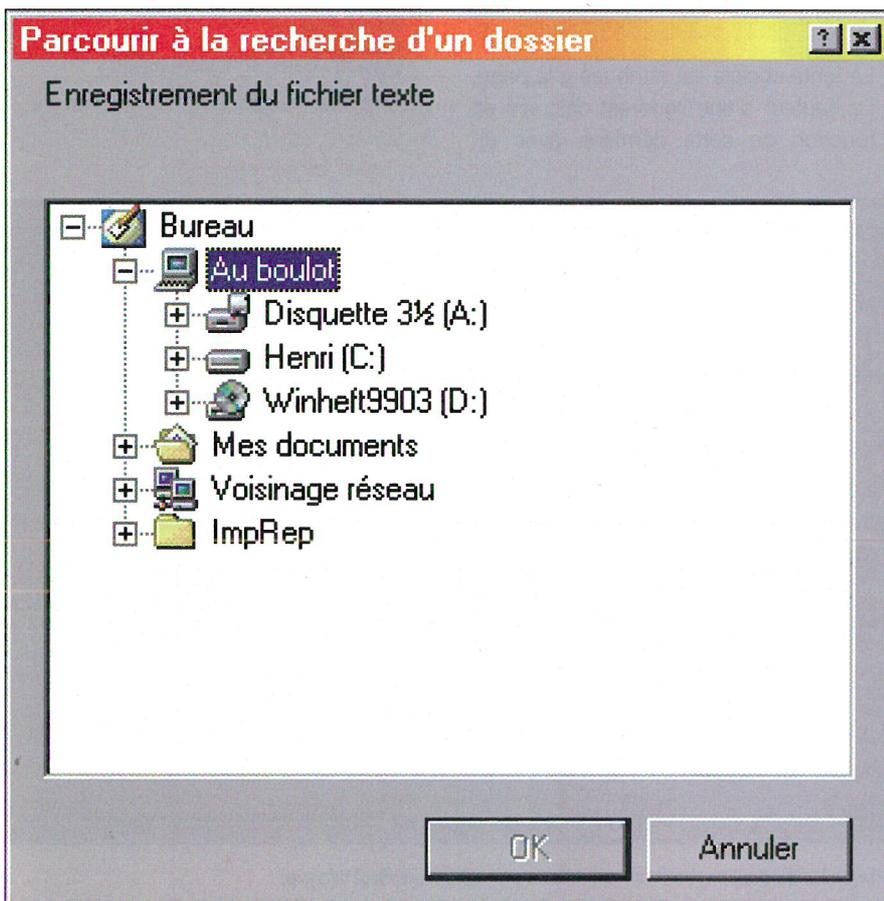
Dim IntHaut As Integer
Dim StrProv As String

IntHaut = LngHauteurPageTwips -
But.TextHeight("Paget 9/9") - Espace
But.CurrentX = IntGauche
But.CurrentY = IntHaut
StrProv = "Page " + Format(PageEnCours) + "/" +
Format(IntNombrePage)
But.CurrentX = (But.Width - TextWidth(StrProv)) \ 2
But.Print (StrProv)
End Sub

```

Ces routines se laissent comprendre par elles-mêmes.

> Figure 1 : l'interface d'exploration classique du poste de travail sous Windows est utilisée par ImpRep.



## Le fond des résultats

L'opération la plus importante de cet utilitaire est sans conteste la recherche du contenu des répertoires et la mémorisation de ces données. La recherche de fichiers ayant été abordée dans un numéro précédent de votre revue préférée, je me contenterai de rappeler les principales fonctions ApiWin32 utilisées ici (**voir tableau 4**).

Une boucle de recherche se composera toujours sur le schéma suivant :

Ouverture de la recherche avec FindFirstFile, à qui l'on passe la chaîne de recherche.

En cas de succès, ne pas oublier de traiter le premier élément trouvé, qui est renvoyé dans FindFileData.

Début de la boucle :

- continuer la recherche avec FindNextFile,
- en cas de succès, traiter l'élément envoyé par FindFileData.

Fin de la boucle lors d'une erreur.

La mémorisation du résultat des recherches se fait dans ListeChemins.bas. Examinez en détail ce fichier et vous verrez que j'ai pris une voie qui peut vous paraître compliquée mais qui m'a semblé la mieux adaptée à mes besoins. Au lieu de choisir une collection existante, j'ai préféré travailler avec un tableau de fiches, utilisant ReDim pour l'agrandir. Une fiche se décompose comme suit : (**voir tableau 8**). Je vous conseille de regarder en détail les fonctions qui ajoutent un chemin ou un fichier. Une ligne peut vous surprendre :

```
ReDim Preserve Fichier_Chemin(Nombre_Chemin)
```

Elle permet d'agrandir le tableau sans perdre les données qui y sont déjà stockées. Sans Preserve, tout serait effacé. Examinons maintenant la procédure la plus mystérieuse de ce fichier :

```

Public Sub calcul_trait()
Dim i As Integer
Dim j As Integer
Dim col As Integer

For i = Nombre_Chemin To 2 Step -1
If Fichier_Chemin(i).Type_fichier = 0 Then
col = Fichier_Chemin(i).Place
For j = i - 1 To 2 Step -1
If Len(Fichier_Chemin(j).Dessin) >=
Len(Fichier_Chemin(i).Dessin) Then
If Mid(Fichier_Chemin(j).Dessin, col, 1) = " "
Then
Mid(Fichier_Chemin(j).Dessin, col, 1) = ""
End If
If Mid(Fichier_Chemin(j).Dessin, col, 1) = "L"
Then

```

```

Mid(Fichier_Chemin(j).Dessin, col, 1) = "}"
End If
Else
Exit For
End If
Next j
End If
Next i
End Sub

```

Cette routine a pour fonction de définir quelle sorte de trait doit se placer avant l'icone du fichier. Pour chaque répertoire contenu dans un autre répertoire, il faut un décalage vers la droite, pour bien montrer la parenté de l'un par rapport à l'autre. Il faut arriver à une structure de ce type :

```

C:\Basic\Programmez \Envoyé Place = 0
Démarrage Place = 1
Article Place = 2

```

En tenant compte de la place de chaque répertoire dans l'arborescence, la routine détermine s'il y a lieu de modifier le signe précédent l'icone. En tenant compte de ces données, les lignes de programmes deviennent plus claires. Ainsi lors du dessin de la ligne sur la page, plus aucun calcul n'est à faire. Il suffit d'interpréter les signes "I", "}" ou "L" pour savoir ce qu'il faut dessiner.

## Les icônes

Revenons justement à l'affichage d'une ligne sur la page affichée à l'écran ou sur l'imprimante :

```

Private Sub DessineLigne(But As Object, num As Integer, haut As Long)
Dim LngX As Long
Dim IntEntre As Integer

IntEntre = EnTwips(10)
But.CurrentX = IntGauche + (IntHauteurLigne * Fichier_Chemin(num).Place)
But.CurrentY = haut
DessineTrait But, num
But.CurrentY = haut + EnTwips(5)

RecupereIcone Fichier_Chemin(num).Nom_Complet, PctFichier
But.PaintPicture
PctFichier.Picture, But.CurrentX, But.CurrentY, IntHauteurLigne - EnTwips(5), IntHauteurLigne - EnTwips(5)
But.CurrentX = But.CurrentX + IntHauteurLigne
LngX = But.CurrentX
If num = 1 Then
But.Print (Fichier_Chemin(num).Nom_Complet)
Else
But.Print (Fichier_Chemin(num).Nom_court)
End If
If Fichier_Chemin(num).Type_fichier = 1 Then
But.CurrentY = haut
But.CurrentX = LngX + Fichier_Chemin(num).Tai + IntEntre + IntEntre
LngX = But.CurrentX
If ChkDate.Value = 1 Then

```

```

But.Print
(Format(FileDateTime(Fichier_Chemin(num).Nom_Complet)))
But.CurrentX = LngX +
But.TextWidth(Format(Now)) + IntEntre + IntEntre
End If
If ChkTaille.Value = 1 Then
But.CurrentY = haut
But.Print
(Format(FileLen(Fichier_Chemin(num).Nom_Complet)))
+ " bytes"
End If
End If
DoEvents
End Sub

```

- La routine définit tout d'abord l'espace qui devra se trouver entre

les différentes colonnes.

- L'endroit où sera écrit la ligne est défini en tenant compte, pour la colonne, du décalage rapporté dans Place. La ligne est celle qui a été transmise dans la variable haut.
- La ligne de parenté est dessinée. Je vous laisse regarder la routine DessineTrait, elle ne représente aucune difficulté.
- Le programme corrige ensuite la ligne où doit s'écrire le reste des données. Elle a été modifiée par DessineTrait.

Tableau 3 : le temps des fichiers

Code	Signification
Type FILETIME	Déclaration
DwLowDateTime As Long	Date et heure du fichier, poids faible
DwHighDateTime As Long	Date et heure du fichier, poids fort
End Type	

Tableau 4 : les attributs et abattis divers du programme

Code	Signification
Type WIN32_FIND_DATA	Déclaration
DwFileAttributes As Long	Attribut du fichier
FtCreationTime As FILETIME	Date de création
FtLastAccessTime As FILETIME	Date de la dernière consultation
FtLastWriteTime As FILETIME	Date de la dernière modification
NFileSizeHigh As Long	Taille du fichier Poids fort
NFileSizeLow As Long	Taille du fichier Poids faible
DwReserved0 As Long	Réservé
DwReserved1 As Long	Réservé
CFileName As String * MAX_PATH	Nom du fichier ou du répertoire
CAAlternate As String * 14	Nom court du fichier s'il existe
End Type	

Tableau 5 : à la recherche des fichiers

Code	Signification
Function FindFirstFile (	Déclaration
ByVal lpFileName As String	Nom recherché, peut être *.*
LpFindFileData As WIN32_FIND_DATA	Variable où sera stockée la réponse
) As Long	Renvoi le handle de recherche ou -1 en cas d'échec

- Il faut ensuite récupérer l'icône associée au fichier ou au dossier. RecupereIcone sera traitée en détail plus loin.
- L'image est enfin dessinée sur la page.
- Il faut déplacer ensuite la colonne d'écriture.
- S'il faut afficher la première ligne, le chemin complet sera utilisé, sinon seul le nom du fichier ou répertoire sera utilisé.
- Le reste de la routine ne concerne plus que les fichiers. Il faut tenir compte des options définies par l'utilisateur, pour afficher ou non la date et la taille du fichier. Vous remarquerez que le champ Tai est utilisé. En ayant retenu le plus long nom de fichier rencontré dans le répertoire courant et en l'ayant mémorisé dans ce champ, il est aisé de l'utiliser pour faire de belles colonnes sans que les données ne se chevauchent.

## Afficher les icônes : un travail de routine

Continuons par la routine RecupereIcone. Cette dernière utilise le type et les routines WinApi32 développés dans le **tableau 4**.

Le but recherché était de pouvoir afficher l'icône du programme ou du répertoire à côté du nom. Pour ce faire, il fallait récupérer l'icône affichée par Windows. Suivons pas à pas cette routine :

```
Public Sub RecupereIcone(StrFichier As String, ByRef
PctBox As PictureBox)

    Dim shfi As SHFILEINFO
    Static Initial As Byte

    If Initial = 0 Then
        Initial = 1
        InitCommonControls
    End If
    PctBox.Cls
    PctBox.Picture = LoadPicture("")

    SHGetFileInfo ByVal StrFichier, 0&, shfi, Len(shfi),
    SHGFI_ICON Or SHGFI_SMALLICON
    DrawIcon PctBox.hdc, 0, 0, shfi.hIcon

    PctBox.Refresh
    Set PctBox.Picture = PctBox.Image
End Sub
```

> La définition de la sorte de trait devant se placer avant l'icône du fichier et le décalage vers la droite permettent de montrer la parenté des fichiers les uns par rapport aux autres.

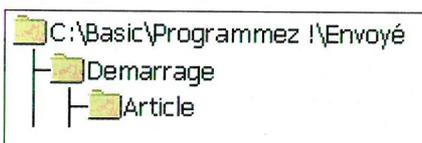


Tableau 6 : premier renvoi après une déclaration

Code	Signification
Function FindNextFile(	Déclaration
ByVal hFindFile As Long	Handle de recherche renvoyé par FindFirstFile
LpFindFileData As WIN32_FIND_DATA	Variable où sera stockée la réponse
) As Long	Renvoi 0 en cas d'échec

Tableau 7 : second renvoi après une déclaration

Code	Signification
Function FindClose(	Déclaration
ByVal hFindFile As Long	Handle de recherche renvoyé par FindFirstFile
) As Long	Renvoi 0 en cas d'échec

Tableau 8 : intégration du WordPad de Windows

Code	Signification
Type TFiche	Déclaration
Nom_Complet As String	Nom complet du fichier, soit le chemin et le nom : C:\Windows\Write.exe
Nom_court As String	Nom du fichier sans le chemin : Write.exe
Place As Integer	Décalage vers la droite en fonction des répertoires parents.
Dessin As String	Indication sur la forme de la ligne qui doit être dessinée devant l'icône du programme
Type_fichier As Byte	Fichier (1) ou répertoire (0).
Tai As Long	Taille du nom le plus grand rencontré dans le répertoire.
End Type	Fin de la déclaration

```
DestroyIcon shfi.hIcon
End Sub
```

- Avant de pouvoir utiliser SHGetFileInfo, il faut initialiser une fois les contrôles. J'ai privilégié la solution de la variable statique. Cette dernière me semblait plus à même de satisfaire mes exigences, c'est-à-dire pouvoir utiliser cette fonction dans n'importe quel programme, sans avoir à y ajouter de modifications.
- La procédure InitCommonControls n'est donc initialisée qu'une seule fois, puisqu'un contrôle se fait au niveau de la variable Initial.
- La PictureBox où sera dessiné l'icône obtenu est effacée. Attention, il est important d'effacer aussi bien la propriété Picture avec LoadPicture() que la

propriété Image avec CLS.

- Les informations sur le fichier ou le chemin sont demandées à Windows. Le flag SHGFI\_SMALLICON, qui fournit le petit icône, a été choisi pour les meilleures qualités graphiques que l'on obtient lors de l'affichage de l'icône. SHGFI\_LARGEICON propose également un dessin d'icône, l'uniformité de ses couleurs est cependant moins bonne.
- L'icône reçu est dessiné sur l'objet PictureBox.
- Windows doit dessiner le résultat.
- Comme cité plus haut, la propriété Image a acquis l'image de l'icône, mais Picture est toujours vierge. Transmettons-lui donc ce dessin.
- Détruisons l'icône fourni pour ne pas épuiser les ressources.

## Pour tirer le meilleur parti du résultat

Il s'agit maintenant de tirer parti de ce que nous avons construit. Si l'utilisateur désire envoyer les pages créées sur l'imprimante, donnons-lui également la possibilité de n'imprimer qu'une partie de ces dernières. La gestion des pages à imprimer est importante mais facile à réaliser. Le nombre de pages à imprimer nous est connu.

Lors du chargement de la feuille de dialogue, les deux champs de saisie peuvent déjà être remplis. Il ne reste plus qu'à effectuer le petit contrôle qui suit lors du changement de valeur de la saisie de la page de départ :

```
Private Sub TxtDepuis_Change()
    ' Contrôle des valeurs entrées
    Dim IntDepuis As Integer ' Page de départ
    Dim IntJusque As Integer ' Dernière page

    IntDepuis = Val(TxtDepuis.Text)
    IntJusque = Val(TxtJusque.Text)
    If IntDepuis < 1 Then IntDepuis = 1
    If IntDepuis > FrmImpRep.IntNombrePage Then
        IntDepuis = FrmImpRep.IntNombrePage
    If IntDepuis > IntJusque Then IntJusque = IntDepuis
    TxtDepuis.Text = IntDepuis
    TxtJusque.Text = IntJusque
End Sub
```

- Le contenu des deux boîtes de saisie est transformé en nombres entiers.
- La première page est-elle plus petite que 1 ?
- La dernière page est-elle plus grande que le nombre de pages à imprimer ?
- La première page est-elle plus petite que la dernière page ?
- Remettons les valeurs en place après modification.

La même méthode de contrôle est utilisée pour le changement de valeur dans le champ de saisie de la dernière page. En cas de pression sur le bouton OK, la fenêtre de dialogue passe la main Imprime de FrmImpRep.

```
Public Sub imprime(Depuis As Integer, Jusque As Integer)
    Dim IntCompteur As Integer
    Dim IntNum As Integer
    IntNum = PageEnCours
    For IntCompteur = Depuis To Jusque
        DessinePage Printer, IntCompteur
    If IntCompteur <> Jusque Then
        Printer.NewPage
    Else
        Printer.EndDoc
    End For
End Sub
```

### Tableau 9 : informations sur le fichier

Code	Signification
Type SHFILEINFO	Information sur un fichier
hIcon As Long	Icône
ilIcon As Long	Index de l'icône
dwAttributes As Long	Flags
szDisplayName As String * MAX_PATH	Nom du fichier
szTypeName As String * 80	Type du fichier
End Type	Fin de déclaration

### Tableau 10 : initiation initiale

Code	Signification
Sub InitCommonControls ()	Initialise

### Tableau 11 : contrôle d'identité

Code	Signification
Function SHGetFileInfo()	Prend les informations sur un fichier
ByVal pszPath As Any,	Nom du fichier
ByVal dwFileAttributes As Long,	Attributs du fichier
psfi As SHFILEINFO,	Fiche de type SHFILEINFO
ByVal cbFileInfo As Long,	Info sur le fichier
ByVal uFlags As Long	Flags
) As Long	

### Tableau 12 : icône sur le périphérique

Code	Signification
Function DrawIcon (	Dessine un icône
ByVal hdc As Long,	Contexte du périphérique où l'on veut dessiner
ByVal x As Long,	Emplacement de l'icône
ByVal y As Long,	Emplacement de l'icône
ByVal hIcon As Long,	Handle de l'icône à dessiner
) As Long	Non zéro en cas de succès

### Tableau 13 : le non zéro du succès

Code	Signification
Sub InitCommonControls ()	InitialiseFunction DestroyIcon( Détruire l'icône
ByVal hIcon As Long	Handle de l'icône à détruire
) As Long	Non zéro en cas de succès

```
End If
Next IntCompteur
PageEnCours = IntNum
End Sub
```

Elle ne fait que peu de choses en fait. Depuis la page de départ à la page de fin, elle redessine les pages sur l'imprimante et envoie l'impression.

Pour la sauvegarde des images, le principe est strictement le même. Le choix des pages à sauvegarder est donné à l'utilisateur, puis chaque page choisie est sauvée sur le disque.

Il ne faut cependant pas oublier que la sauvegarde d'une image sur le disque suit, en Visual Basic, une règle un peu spéciale : la bitmap sauvée aura une résolution identique à celle de l'écran. Cette règle peut produire des fichiers extrêmement volumineux. Seul le choix du répertoire est ajouté. Il utilise une méthode déjà abordée précédemment. La déclaration de cette dernière est :

```
Public Declare Function SHGetPathFromIDLList Lib
"shell32.dll" Alias "SHGetPathFromIDLListA" (ByVal
pidl As Long, ByVal pszPath As String) As Long
```

Elle est basée sur une variable de type BROWSEINFO dont les principaux champs sont développés dans le **tableau 14**. Au retour, il ne reste plus qu'à récupérer le chemin choisi avec :

```
Public Declare Function SHBrowseForFolder Lib
"shell32.dll" Alias "SHBrowseForFolderA"
(lpBrowseInfo As BROWSEINFO) As Long
```

> Figure 2 : l'écran configurant vos préférences en matière de répertoires, polices et couleurs.

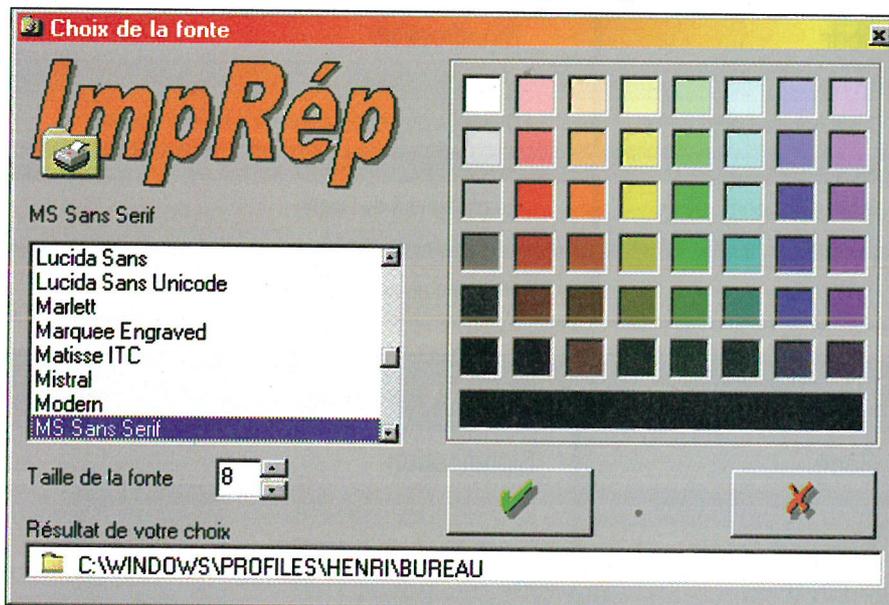


Tableau 14 : parent par la fenêtre

Code	Signification
Type BROWSEINFO	
hOwner As Long	Fenêtre parent
pidlRoot As Long	
pszDisplayName As String	Ligne à afficher
lpszTitle As String	Chaîne de recherche
lpfn As Long	
lParam As Long	
image As Long	
End Type	

Ces routines vous donneront la fenêtre de dialogue de la **figure 1** page 72.

### Mais que fait la police ?

Le choix de la fonte et de la couleur du texte se présente ainsi : voir **figure 2** ci-dessous.

Vous avez maintenant tous les éléments en main pour comprendre chaque ligne de cette fenêtre de dialogue. Lors du chargement de la feuille, les fontes disponibles sont chargées et les valeurs en cours sont prises dans FrmImpRep. Lors de la modification du choix de la fonte ou de sa taille, l'exemple en bas de fenêtre est adapté.

En cas de choix positif, les valeurs modifiées sont retransmises à FrmImpRep.

Vous avez là un bon point de départ pour faire votre propre utilitaire d'impression de répertoires. Vu la facilité de la programmation d'un aperçu avant impression, n'hésitez surtout pas à vous inspirer de ce petit exemple pour améliorer vos logiciels.

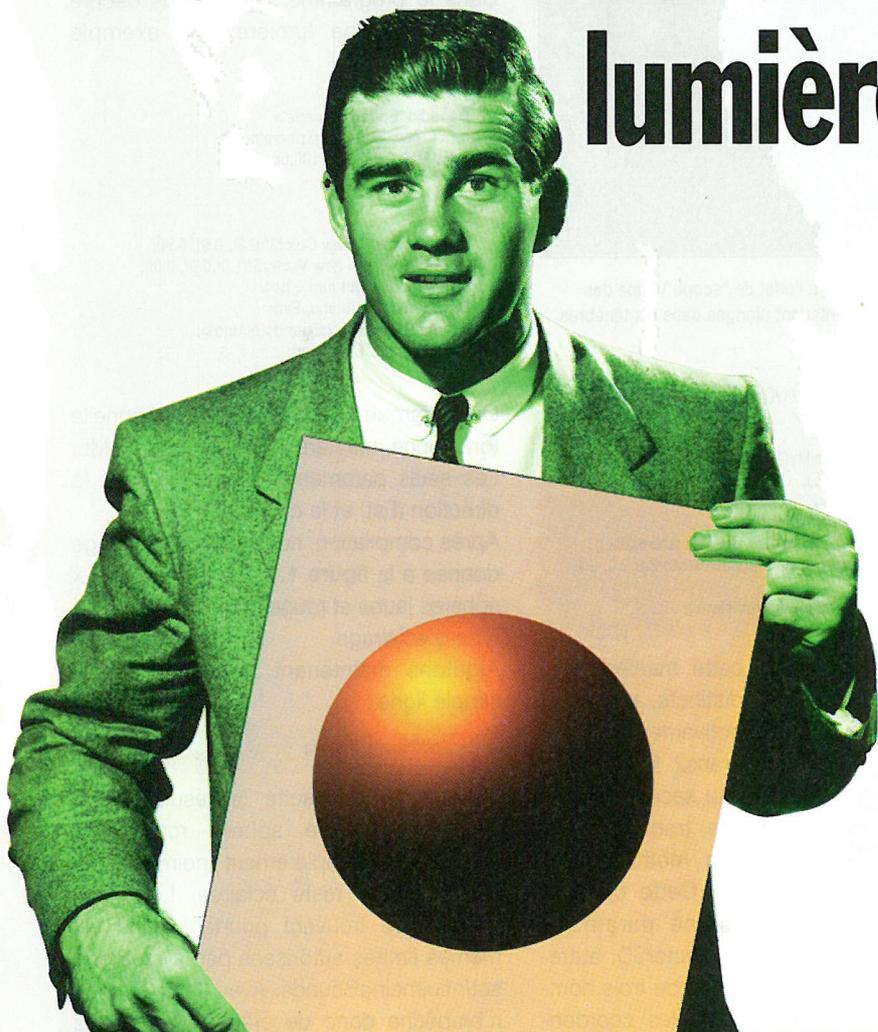
Ainsi, Henri Ruch, notre Peter Norton transalpin fanatique de VB, ne cesse de fournir des solutions techniques astucieuses à des besoins quotidiens, auquel nul n'échappe.

Retrouvez les sources et binaires de cet article, prêts à l'emploi sur notre CD-ROM, dans le répertoire REDAC. Bien entendu, vous pouvez utiliser le logiciel ImpRép pour dresser l'inventaire des listings dont vous disposez depuis le premier numéro de *Programmez !*. ■



> Les deux derniers ouvrages pratiques sur le langage Visual Basic édités par Microsoft Press.

# Le siècle des lumières virtuelles



Java 3D et VRML, deux langages a priori opposés dans leur philosophie mais qui seront finalement qualifiés de cousins, compte tenu de leurs points communs. Si le premier gère la lumière avec une souplesse exemplaire, le second autorise plus de personnalisation. Voici quelques astuces pour le siècle de la 3D.

■ par Rémi Sussan  
sremi@compuserve.com

## 1 Java 3D

Java 3D s'avérant, à l'instar du VRML, un système avant tout réservé aux applications temps réel, on ne s'étonnera pas des nombreuses ressemblances existant entre ces deux technologies. Java 3D s'inspire largement d'OpenGL, produit par Silicon Graphics qui conçoit la norme VRML 2. On peut donc considérer les deux systèmes comme cousins.

Contrairement à VRML cependant, Java 3D met à disposition du programmeur une "ambientLight" dénuée d'origine, capable d'éclairer les divers objets de la scène de

manière uniforme. Il est donc beaucoup plus simple de spécifier la lumière ambiante en Java 3D qu'en VRML, quoiqu'on perde du coup une source possible d'effets spéciaux.

Pour spécifier une lumière ambiante :

```
Color3f grisblanc= new Color3f(0.5f;0.5f;0.5f);
AmbientLight amb = new AmbientLight(grisblanc);
amb.setInfluencingBounds(mybounds);
objRoot.addChild(amb);
```

On constate qu'on ne peut décrire l'intensité de la lumière ambiante. On travaille donc sur sa couleur, ici par

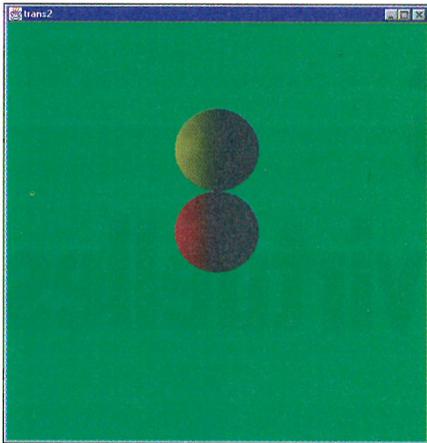
exemple en n'optant pas pour un blanc trop puissant par les paramètres (1.0f;1.0f;1.0f).

Chaque lumière agit dans les limites d'une sphère virtuelle, décrite par l'instruction :

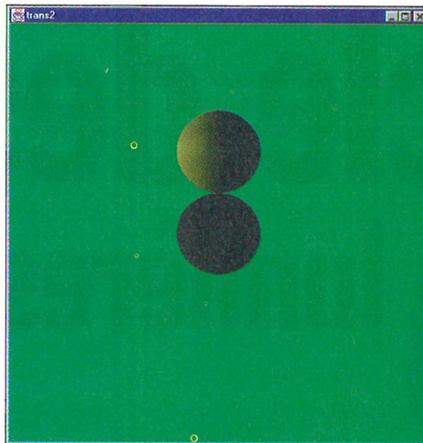
```
amb.setInfluencingBounds(mybounds);
```

Auparavant, on aura toutefois spécifié la variable mybounds par :

```
BoundingSphere mybounds = new
BoundingSphere(
new Point3d( 0.0, 0.0, 0.0
1000.0 );
```



> Figure 1 : deux sphères sont éclairées par une même lumière directionnelle.



> Figure 2 : grâce à l'effet de "scope", l'une des sphères est maintenant plongée dans les ténèbres.

Cela signifie donc que la lumière agit dans une sphère de 1 000 unités, centrée autour de l'origine.

Si cette notion de "bounds" (limites) s'avère assez simple à saisir, Java 3D utilise aussi un autre système, qui représente une source très intéressante d'effets spéciaux : le "scope" ou la "portée".

Java 3D repose en effet sur un "graphe de scène", une arborescence dans laquelle chaque objet est "enfant" d'un élément supérieur. Cette structure est également présente en VRML ou dans POV-RAY, mais de manière beaucoup moins rigide qu'en Java 3D. Or, c'est précisément cette notion de "graphe de scène" qui permet l'application de la notion de portée. Les différents objets de la scène appartiennent à des groupes très précisément définis. Rien n'interdit alors de demander à une lumière d'éclairer une forme, tout en laissant dans les ténèbres une autre située juste à côté, parce que les deux objets ne sont pas membres du même groupe, de la même branche de l'arborescence.

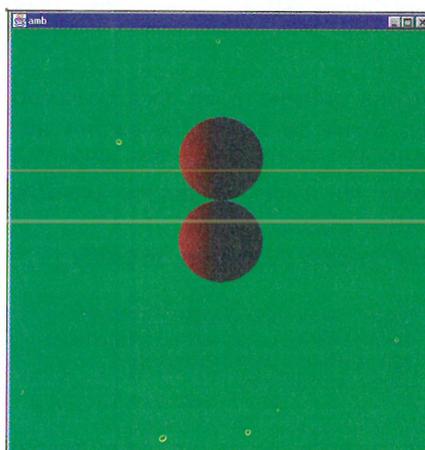
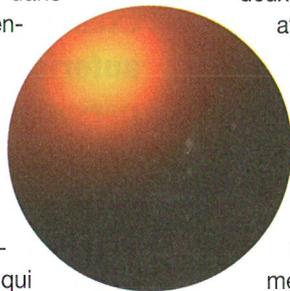
Construisons ainsi un univers virtuel comprenant deux sphères, chacune appartenant à un groupe "transform" différent :

```
TransformGroup objTrans = new
TransformGroup();
TransformGroup Trans2 = new
TransformGroup();
```

```
objTrans.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
```

```
Trans2.setCapability(TransformGroup.ALLOW_TRANSFORM_WRITE);
objRoot.addChild(Trans2);
objRoot.addChild(objTrans);
Transform3D translate= new Transform3D();
translate.setTranslation( new Vector3d( 0.0, 0.4, 0.0 ));
Trans2.setTransform( translate );
```

Nous avons de cette manière créé deux groupes distincts, que nous avons respectivement baptisé Trans1 et Trans2. Nous avons appliqué au second groupe une petite translation en utilisant la méthode setTransform. Cette dernière prend comme paramètre une variable Vector3D, autrement dit une suite de trois nombres qui représentent les coordonnées x, y et z du système cartésien.



> Figure 3 : deux sphères apparemment identiques, bien que leur couleur ambiante soit différente.

Ensuite, on génère deux sphères :

```
objTrans.addChild(new Sphere(0.2f,Apparence));
Trans2.addChild(new Sphere(0.2f,App));
```

Dans le programme, nous allons décrire également une lumière, par exemple directionnelle :

```
BoundingSphere bounds =
new BoundingSphere(new
Point3d(0.0,0.0,0.0), 100.0);
```

...

```
Color3f blanc = new Color3f(0.9f, 0.9f, 0.9f);
Vector3f Est = new Vector3f(1.0f,0.0f, 0.0f);
DirectionalLight lum = new
DirectionalLight(blanc, Est);
lum.setInfluencingBounds(bounds);
objRoot.addChild(lum);
```

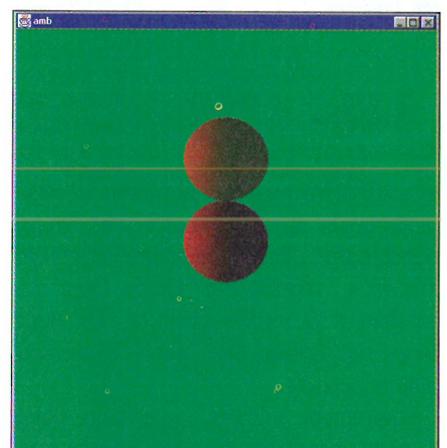
On remarque que la lumière directionnelle fonctionne exactement comme en VRML. Les seuls paramètres utilisés ici sont la direction (Est) et la couleur.

Après compilation, nous obtenons l'image donnée à la figure 1. On le voit, les deux sphères jaune et rouge subissent le même type d'éclairage.

Ajoutons maintenant à notre code la simple ligne :

```
lum.addScope(Trans2);
```

Et compilons ensuite le résultat : on constate que la sphère rouge est maintenant complètement noire, tandis que la jaune reste éclairée. Les deux sphères se trouvent pourtant dans les mêmes limites autorisées par la méthode setInfluencingBounds. Avec Java 3D, rien n'empêche donc de créer des lumières qui n'éclaireront que certains groupes



> Figure 4 : lorsqu'on ajoute une lumière ambiante à la scène, la différence entre les 2 sphères apparaît.

d'objets spécifiés et ce indépendamment de leur position dans l'espace.

## Les couleurs Java

Java 3D s'avère être plus complexe, dans ce domaine, que VRML ou POV. En effet, il ne faut pas moins de quatre couleurs pour définir un matériau complet. On dispose donc d'une couleur diffuse, spéculaire, émissive mais également d'une couleur ambiante, en lieu et place d'une simple valeur d'intensité.

Contrairement encore au VRML, les sources lumineuses Java ne possèdent pas de paramètre "ambientIntensity" à partir duquel les parties dans l'ombre peuvent néanmoins recevoir une petite quantité de lumière. C'est donc avec la "lumière ambiante" sans origine, présente dans tout l'univers virtuel, que la "couleur ambiante" d'un objet pourra réagir.

Pour comprendre l'usage de ce nouveau paramètre, reprenons le programme

précédent, supprimons la ligne se rapportant au "scope", puis assignons à nos deux sphères un matériau quasi identique.

```
Color3f jaune = new Color3f(0.8f,0.8f,0.0f);
Color3f rouge = new Color3f(0.80f, 0.20f, 0.2f);
Color3f noir = new Color3f(0.0f,0.0f,0.0f);

Material Matiere =
    new Material(rouge,noir , rouge, rouge,
110.f);
Appearance Apparence = new Appearance();
Apparence.setMaterial(Matiere);
Material Materiel =
    new Material(jaune,noir , rouge, rouge,
110.f);
Appearance App = new Appearance();
App.setMaterial(Materiel);
....

Trans.addChild(new Sphere(0.2f,Apparence));
Trans2.addChild(new Sphere(0.2f,App));
```

La classe "Material" contient cinq paramètres : la couleur ambiante, émissive (dans notre cas noire, c'est-à-dire inexistante), diffuse et spéculaire. La valeur numérique finale est la brillance. Elle fonctionne comme en VRML.

Entre "Materiel" et "Matière", il y a une seule différence : la couleur ambiante, rouge ou jaune.

N'oublions pas que le programme comporte une unique source lumineuse, qui est de type directionnel. Effectuons une compilation du fichier et observons maintenant le résultat tel qu'il apparaît en **figure 3**. Les deux sphères sont rigoureusement identiques.

Ajoutons ensuite au monde virtuel une lumière ambiante (ici blanche) par les instructions suivantes :

```
Color3f blancfaible = new Color3f(0.2f, 0.2f, 0.2f);
AmbientLight amblight = new
AmbientLight(blancfaible);
alLt.setInfluencingBounds(mybounds);
objRoot.addChild(amblight);
```

Après une nouvelle compilation, nous constatons que la sphère la plus haute a pris une teinte légèrement jaune. De plus, les hémisphères ombrés sont maintenant plus éclairés que précédemment. ■

## 2 VRML 97

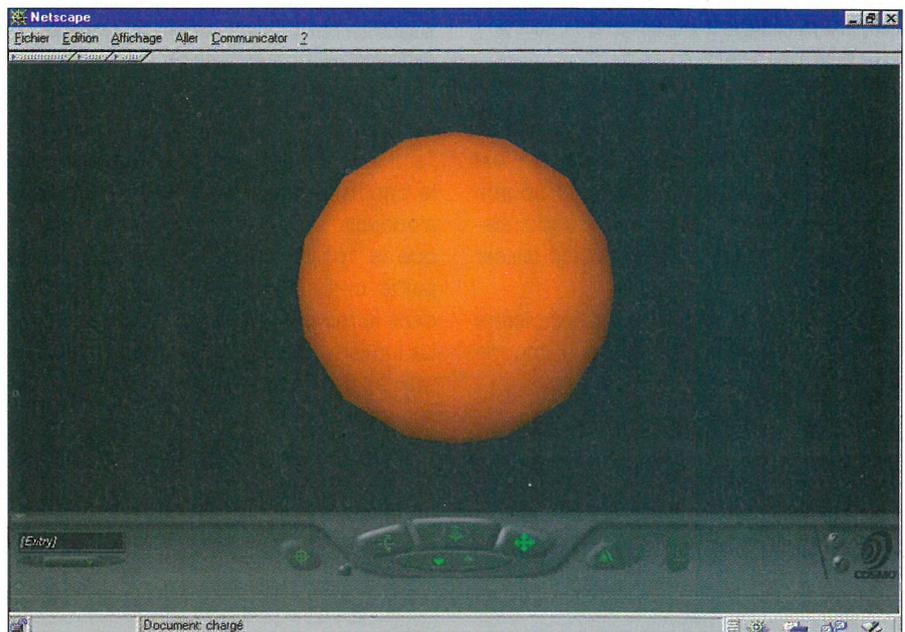
On dispose en VRML de trois catégories de sources lumineuses : ponctuelle, directionnelle et spots. A ces lumières, créées par l'auteur du programme, s'ajoute la "headlight". Cette lampe, installée automatiquement par le navigateur, illumine systématiquement tout objet se trouvant devant le visiteur. Cette lumière supplémentaire s'avère précise dans certains cas, gênante dans d'autres. L'internaute peut l'activer ou la supprimer à partir de son navigateur. Il est également permis au programmeur de désactiver son usage grâce à un paramètre du nœud NavigationInfo :

```
NavigationInfo {
    headlight FALSE
}
```

La lumière ponctuelle exige avant tout une "location" (défaut : 0,0,0), un "rayon" (défaut : 100), une intensité (défaut : 1) et une couleur (défaut : blanc). A ces paramètres de base s'ajoutent d'autres, moins fréquents, à savoir "ambientIntensity" et "attenuation".

Notons tout d'abord la différence entre le VRML et un système comme POV : la lumière ponctuelle possède avec le premier un rayon, c'est-à-dire une limite à sa portée. Contrairement au ray-tracer, celle-ci n'est pas infinie.

Le paramètre "intensity" indique la puissance de la source, il varie entre 1 et 0. Le paramètre "attenuation" décrit comment cette "intensity" diminue entre le centre du cercle défini par le rayon et sa périphérie. Malheureusement, il est quasi



> Une sphère orange simple, basée uniquement sur un procédé de couleur diffuse.

impossible de spécifier les trois paramètres composant cette atténuation de manière intuitive. Ils forment en effet une complexe équation de type :

$$1/\max(\text{attenuation}[0] + \text{attenuation}[1]x + \text{attenuation}[2]x^2, 1)$$

On ne peut que conseiller de procéder par essais et erreurs ! Il est bon toutefois de savoir que les valeurs de ces trois paramètres doivent être supérieures ou égales à zéro.

L'ambientIntensity est beaucoup plus facile à saisir dès que l'on comprend le rôle que joue la lumière ambiante en VRML. Comme dans POV, celle-ci permet d'éclairer quelque peu les portions d'un objet caché par rapport aux sources lumineuses, en évitant de cette manière qu'il disparaisse complètement dans les

```
Material{diffuseColor 1.0 0.5 0.0}
  geometry Sphere{
  }
```

```
PointLight{
  location -10,0,0
  radius 11
  #ambientIntensity 1
}
```

Le paramètre "ambientIntensity" ne figure dans ce listing qu'en commentaire. Autrement dit, "ambientIntensity" n'existe pas pour le navigateur VRML. Lorsque vous chargez ce fichier dans Cosmo Player ou Worldview, vous pouvez voir une sphère orange qui est éclairée d'un seul côté. La partie située dans l'ombre est quasiment invisible.

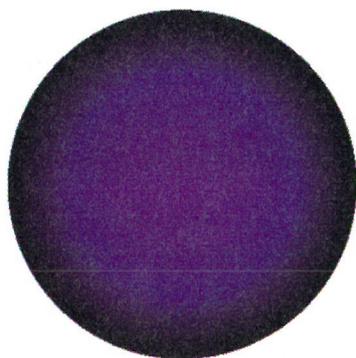
Si nous supprimons maintenant le

Vous pouvez remarquer l'absence de paramètres "location" et "radius". La lumière directionnelle est de portée infinie, elle ne possède pas d'origine. Comme nous l'avons indiqué précédemment, elle sert à simuler des sources lumineuses très puissantes et lointaines, comme le soleil par exemple.

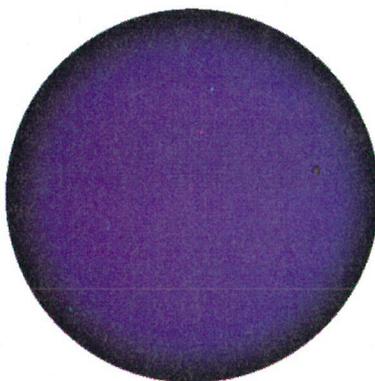
```
DirectionalLight{
  ambientIntensity 0.8
  direction 1 0 0
}
```

Indique de cette manière que la lumière vient du côté gauche et se dirige vers la droite, c'est-à-dire vers les valeurs positives de l'axe des x, comme le montre le chiffre "1" en première position.

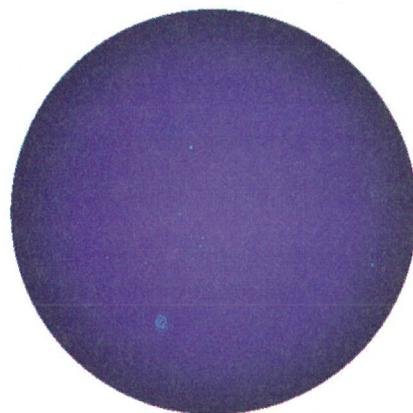
De la même manière :



> Le double cône d'un spot : ici, sous POV-RAY, le cône interne possède une unité de rayon, mais le cône externe en aura 10.



> Ici, le cône interne est plus large (5 unités). Pour augmenter l'effet, on a également fait passer le paramètre tightness de 100 à 50.



> Dans cette dernière image, les cônes internes et externes sont de taille équivalente.

ténèbres. Mais en VRML, il n'existe pas de paramètre "ambient\_light", comme c'est le cas dans POV. Chaque source lumineuse doit donc éclairer (paradoxalement) les éléments qu'elle devrait laisser dans l'ombre.

Pour comprendre la logique assez bizarre de ce système, saisissons un monde virtuel très simple :

```
#VRML V2.0 utf8
NavigationInfo {
  headlight FALSE
}

Shape{
  appearance Appearance {material
```

caractère # et si nous activons "ambientIntensity" : on peut constater que le contraste entre les deux hémisphères s'adoucit. Ce phénomène est dû au fait que la "pointlight" illumine légèrement la partie ombrée, ce qui serait impossible dans le monde réel.

La lumière directionnelle s'exprime, dans les spécifications, par la formule suivante :

```
DirectionalLight {
  exposedField SFFloat ambientIntensity 0 # [0,1]
  exposedField SFColor color 1 1 1 # [0,1]
  exposedField SFVec3f direction 0 0 -1 # (-)
  exposedField SFFloat intensity 1 # [0,1]
  exposedField SFBool on TRUE
}
```

```
DirectionalLight{
  ambientIntensity 0.8
  direction 0 -1 0
}
```

Ce qui signifie que la lumière vient du haut et se dirige vers le bas, autrement dit, vers les valeurs négatives de l'axe des y (-1 en seconde position).

Reste à définir la syntaxe des spots :

```
SpotLight {
  exposedField SFFloat ambientIntensity 0 # [0,1]
  exposedField SFVec3f attenuation 1 0 0 # [0,]
  exposedField SFFloat beamWidth 1.570796 # (0,2]
  exposedField SFColor color 1 1 1 # [0,1]
  exposedField SFFloat cutOffAngle 0.785398 # (0,2]
  exposedField SFVec3f direction 0 0 -1 # (-)
```

```

exposedField SFFloat intensity 1 # [0,1]
exposedField SFVec3f location 0 0 0 # (-,
exposedField SFBool on TRUE
exposedField SFFloat radius 100 # [0,
}

```

Parmi tous ces paramètres, seuls importent réellement la location, le "beamWidth" (largeur du rayon), le "cutOffAngle", la direction et le radius (rayon). Parmi eux, ce sont bien évidemment "beamWidth" et "cutOffAngle" qui déterminent les caractères propres du spot.

Le cône externe est spécifié par "cutOffAngle", le cône interne par "beamWidth". Logiquement, "beamWidth" devrait toujours posséder une valeur inférieure à celle de "cutOffAngle" mais, en fait, vous pouvez en spécifier une supérieure. Dans ce cas, le cône externe ne subit aucune déperdition et la luminosité reste uniforme sur tout le volume couvert. Attention, on décrit ces cônes en spécifiant leur degré d'ouverture en radians, mesure peu intuitive mais préférée aux degrés par le VRML, parce que d'une plus grande facilité de traitement par les navigateurs. La dernière couleur, dite émissive, sert à représenter des objets lumineux. Il est conseillé, lors de l'utilisation de ce paramètre, de placer la diffuseColor à 0.



```

Material{
diffuseColor 0 0 0
emissiveColor 0.5 0.5 0.0
}

```

Comme nous l'avons précisé précédemment, un tel objet n'éclairera pas son entourage ! Mais rien n'empêche de placer à l'intérieur de la forme une lumière ponctuelle pour obtenir ce résultat.

## Systemes de couleurs

Lorsqu'on n'utilise pas de placage de textures, le nœud Material permet de donner un aspect sympathique à ses formes tridimensionnelles. Grâce à lui, on peut contrôler la couleur diffuse, spéculaire et ambiante d'un objet. Quoi, encore la couleur ambiante ! Ne venons-nous pas de la traiter avec les sources lumineuses ? En fait, le paramètre "ambientIntensity" du nœud Material sert à moduler la quantité de lumière ambiante reçue par une lumière.

Reprenons le code rédigé plus haut, mais complétons la description de la sphère :

```

Shape{
appearance Appearance {material
Material{diffuseColor 1.0 0.5 0.0
ambientIntensity 0}}
geometry Sphere{
}
}

```

Lorsqu'on ouvre le fichier VRML dans le navigateur, on s'aperçoit que la lumière ambiante émise par la source lumineuse a été modifiée par le paramètre inclus dans la description de sphère.

## Modulations de lumières

Par rapport à une lumière ambiante généralisée à toute la scène, VRML offre donc un bon moyen de moduler l'effet, en indiquant des "ambientIntensity" différentes pour chaque source lumineuse et pour chaque objet.

Nous avons décrit la couleur spéculaire et son fonctionnement dans POV. En VRML, la description est un peu plus complexe, puisqu'au lieu d'une simple valeur numérique indiquant la puissance de ce facteur, il faut choisir une couleur spécifique. Cela implique un réglage plus complexe, mais constitue également une source amusante d'effets spéciaux. Par exemple, ce code :

```

Shape{
appearance Appearance {material
Material{diffuseColor 1.0 0.5 0.0
specularColor 0 1.0 0.2}}
geometry Sphere{
}
}

```

Il montre comment une lumière blanche frappant une sphère orange peut donner naissance à... un reflet vert !

En général, on se contente cependant de définir une teinte proche de la couleur diffuse, mais plus claire, comme :

```
specularColor 1.0 0.7 0.0}}
```

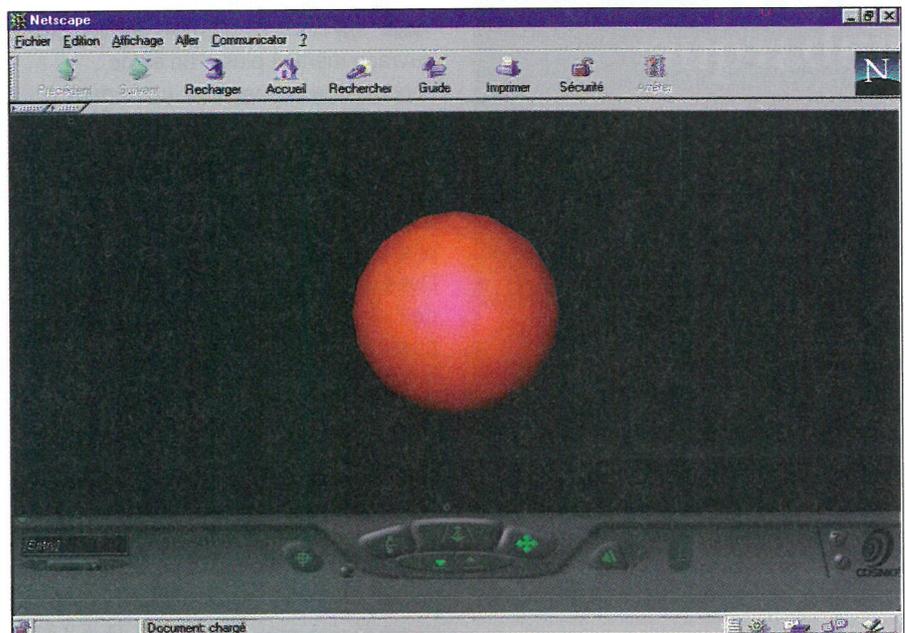
Ou encore comme un niveau de gris très élevé, proche du blanc :

```
specularColor 0.8 0.8 0.8}}
```

Ce qui, comme nous l'avons indiqué plus haut, donnera au volume l'apparence du plastique.

Paramètre supplémentaire, la brillance ou "shininess" permet de contrôler la taille des reflets dus à la spécularité. L'augmentation de cette valeur tend à réduire la taille des reflets obtenus.

La couleur émissive, nous l'avons dit, n'est guère spectaculaire, puisqu'elle tend à aplatir complètement l'objet. Elle sert essentiellement à donner à un objet l'apparence d'une source lumineuse. ■



> Cette sphère orange sous une lumière blanche produit un reflet violet dû à sa couleur spéculaire.

## La bible du programmeur C / C++

Eyrolles, 1 000 pages  
298 F

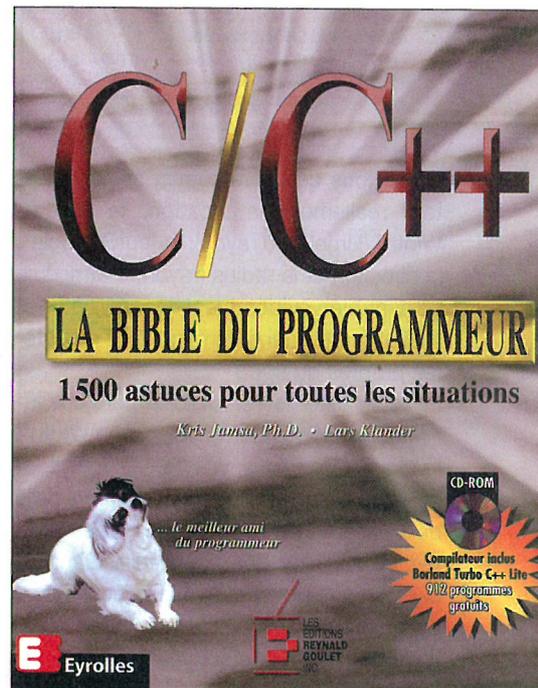
**P**lus grand et plus épais qu'un annuaire de France Telecom, cet ouvrage a été déniché au Canada par les éditions Eyrolles. Il faut croire que ce format hors-norme a fait un effet bœuf aux programmeurs.

La promesse de "1 500 astuces pour toutes les situations" est tenue. Cette compilation a même des airs de newsgroups (groupes de discussion sur Internet). Il y a cette même sensation de véridique, avec questions et réponses du terrain. Ce vivier de solutions est accompagné de références discrètes et synthétiques à la fois sur le langage C et C++. Ces astuces numérotées sont assez courtes car très ciblées. L'essentiel des tips représente une demi-page, voire de temps en temps une page. Il est difficile d'imaginer qu'une de vos erreurs

ou une difficulté C ou C++ puisse passer à travers les mailles de ce missel de plus d'un kilo et demi. A moins que vous ne soyez sous le coup d'un bogue lié à telle ou telle version d'outils de développement.

Pour ceux qui veulent se lancer dans la programmation à partir de ce livre, il faut savoir que le Borland Turbo C++ Lite et environ 900 programmes gratuits sont présent sur le CD d'accompagnement. Il n'est pas sûr que l'ouvrage soit le meilleur pour l'initiation. Kernigan et Ritchie, Bjarne Stroustrup sont mieux placés pour écrire des Bibles sur les langages qu'ils ont créés.

Cet ouvrage serait plutôt une référence bien classée à l'usage des praticiens, une sorte de Vidal pour programmeurs.



## Enfin traduits par Eyrolles...

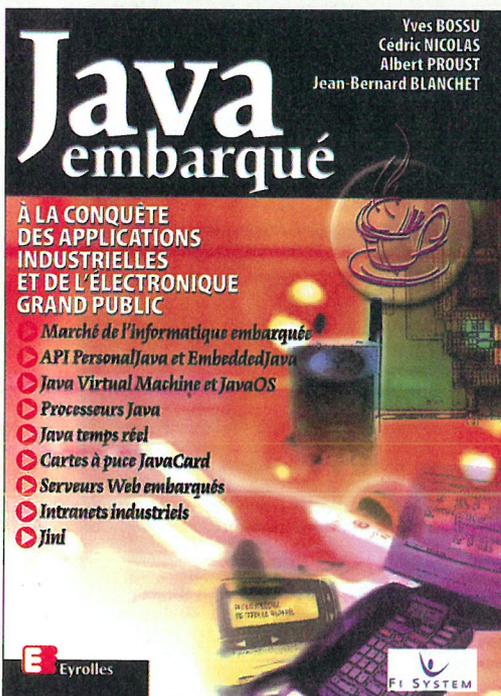
CSS2 (250 pages, 210 F),  
Le développeur Java 2  
(Edition 1999, 890 pages)  
et la fameuse collection  
Wrox comprenant  
notamment Active Server  
Pages 2 (970 pages,  
350 F), Visual C++ 6  
(1 260 pages, 298 F).

à savoir PersonalJava pour le grand public, EmbeddedJava pour l'industrie et JavaCard pour les cartes à puce. Par ailleurs les variantes du Java OS, PicoJava et l'embedded Server, seront expliquées. C'est finalement par une description des applications Java embarquées que les auteurs plaident la cause de ce langage. On y parle d'embarqué : serveur web, carte à puce, network computers, décodeurs web/TV, assistants personnels, bornes interactives et autres applications industrielles, mais avec très peu de listings.

## Java embarqué

Eyrolles, 398 pages  
230 F

**S**orti en même temps que notre dossier *Embarquez vos applications* du mois dernier, cet ouvrage méritait plus d'un éloge : sérieux, complémentaire, up to date. C'est l'un des seuls à traiter Java sous cet angle, avec la participation de trois auteurs de la société Fi System et la consultation de praticiens dans les plus grandes sociétés françaises. Dans un premier temps, c'est le marché de l'informatique embarquée qui est passé en revue, avec les différents modèles d'applications, les mémoires spécifiques et autres contraintes particulières. Le parti pris pour Java est affirmé dans le titre et cela expliquera les jugements des auteurs, parlant d'artisanat novateur pour désigner le marché traditionnel de l'embarqué. Dans la seconde partie, vous découvrirez la diversité des implémentations Java. Les API spécifiques de Java seront détaillées,



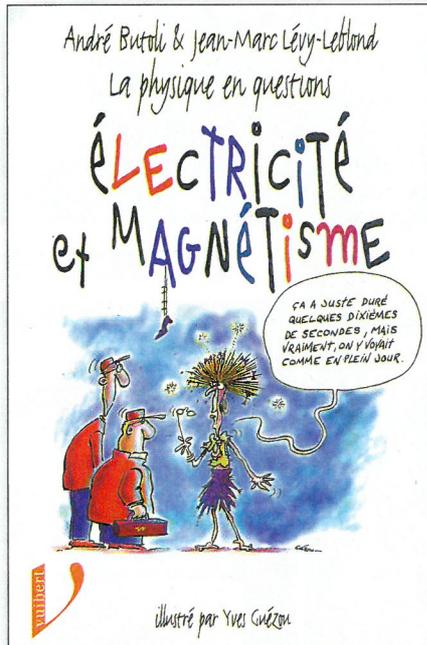
## Collection La physique en questions

Electricité et magnétisme, Mécanique

Vuibert, 130 pages

99 F

La couverture plutôt marrante cache un contenu plutôt sérieux, convenant aux universitaires de première année et aux éternels curieux souhaitant lever le voile sur les forces qui régissent le hardware informatique. Avec le renfort d'illustrations, d'éléments historiques et d'exemples didactiques, on peut se permettre de zapper ce livre de sujet en sujet sans suivre un ordre prédéterminé. Comportant plus de 180 questions, l'ouvrage reprend toutes les connaissances élémentaires de l'électricité et du magnétisme. De quoi parfaire votre culture générale et scientifique. Le premier ouvrage de la collection est toujours disponible et porte sur les sciences mécaniques.



## Les plus belles formules mathématiques

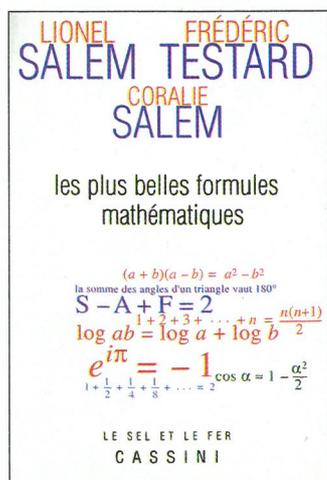
Editions le Sel et le Fer, Cassini, 150 pages

48 F

De la beauté des formules ? Un débat d'esthètes qui peut concerner le programmeur, sans cesse en quête de plus belles implémentations ! Cet ouvrage rejoint le concept du précédent (Electricité et magnétisme, la physique en questions, éditions Vuibert). C'est un livre scientifique illustré de façon conviviale et didactique, pour les amateurs de

mathématiques et même les mauvais élèves qui souhaitent se rattraper sans trop de souffrances.

Les auteurs avouent être à la recherche d'une certaine "harmonie mathématique" en observant la "simplicité des énoncés", "la plastique des symboles" ou "l'évocation esthétique de leur contenu". Mais ces considérations intellectuelles laissent une place au côté ludique des maths. On explique la puissance d'un nombre avec des parts de gâteau ou des arbres généalogiques. Les formes géométriques simples sont décrites comme les plans d'un potager et ainsi de suite pour les angles, équations, logarithmes, exponentielles, suites, volumes, nombres entiers et premiers, le hasard... La conclusion des auteurs met en évidence que la notion d'infini reste la principale aventure des mathématiques, une source de richesses et de découvertes.

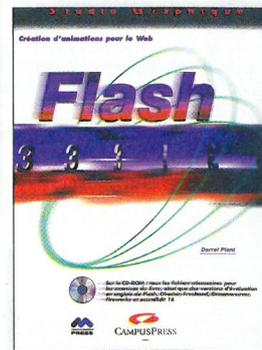


## Flash 3

Campus Press, 360 pages

129 F

Entièrement dédié au logiciel de création graphique pour site web, cet ouvrage est une référence doublée d'un gisement d'astuces pour expérimenter les graphiques vectoriels, les animations et le multimédia en ligne. Pour concevoir l'interactivité des graphiques, un grand passage traitera des Scènes et Actions, qui constituent le mode principal de programmation, ici événementielle et peu contraignante. La principale difficulté de procédés comme



Flash est de savoir penser et créer des "animations légères", pour qu'elles ne soient pas pénalisantes sur le réseau, et l'ouvrage y répond largement.

## Parus chez Microsoft Press...

Dans la collection Guide du programmeur, Visual Interdev 6.0 (330 pages, 279 F), Visual Fox Pro 6.0 (1 000 pages, 389 F), Visual Basic 6.0 (1 000 pages, 379 F). Microsoft Press publie également Visual Basic 6.0 étape par étape (630 pages, 289 F), Atelier Visual Basic 6.0 (800 pages, 349 F), Programmer des composants avec Visual Basic 6.0 (430 pages, 349 F), Dans les coulisses du registre Windows 98 (500 pages, 289 F) et enfin, un collector compte-tenu du contexte juridique de Java : Visual J++ 6.0 en action (190 pages, 159 F).

## CONTACTS DE LA REDACTION

Index alphabétique des sociétés les plus citées depuis le numéro 1 de *Programmez !*

Contactez la rédaction pour obtenir les coordonnées.

Tél : 33 1 43 98 22 22 fax : 33 1 43 28 72 12 e-mail : [feedback@planetepc.fr](mailto:feedback@planetepc.fr)

Société	Spécialités
3COM	Assistant de poche
4X TECHNOLOGIES	Moteur 3D
A2I	Indépendants en informatique
ABERDEEN GROUP	Cabinet d'étude
ACI	SGBD
ACTUATE	XML
ADALOG	ADA
ADDE	Cartographie
ADOBE	Infographie
AFAS	Auteurs de Sharewares
AFNIC	Noms de domaines
AFUL	Utilisateurs de logiciels libres
AFUNT	Utilisateurs de NT
AFUVB	Utilisateurs de VB
ALADDIN Sofware France	Dongles, cartes à puce
ALCATEL	CTI
ALLAIRE	Middleware
ANPE	Emploi
ANVAR	Innovation
ANIX	SDK, modélisation
APCT	Network Computer
APPLE COMPUTER	Macintosh
APR	Recrutement
APRIL	Promotion logiciels libres
ARDENT SOFTWARE	SGBD objet, XML
ARIBA	XML
AT&T	Bjarn Stroustrup
AWARE	SDK, Web
BE EUROPE	Système
BEA SYSTEMS	Middleware, XML
BORLAND	SDK, CORBA
BSA	Piratage
CAMPUS PRESS France	Livres
CASIO	Assistant de poche
CFCE	Commerce extérieur
CHRYSAL SOFTWARE	GED
CIGREF	Grandes entreprises
CLUB JAVA	Java
CNAM	Arts et métiers
COMEDIA	Compilations CD-ROM
COMPUTER ASSOCIATES	Outils, SDK, web, SGBD
COMPUTER Science Corporation	SSII
COMPUWARE FRANCE	Outils SDK
CONTINUUS	Gestion de projet
COREL	Network Computer
COTE D'Azur Développement	Sophia Antipolis
CSE SYSTEMS	Workflow
CUTTER CONSORTIUM	Cabinet d'étude
CYGNUS	Système, logiciels libres
DASAR	Salons
DATACHANNEL	XML
EASDAQ	Finances Euro
ECOO	Système
EDITIONS DUNOD	Livres
EDITIONS EYROLLES	Livres
EDITIONS J-C LATTES	Livres
EDITIONS MASSON	Livres
EDITIONS VUIBERT	Livres
EISTI	Ingénieurs
ELEA CARDWARE	Carte à puce

Société	Spécialités
ENST	Ingénieurs
EPITA	Ingénieurs
FIDJIT	Emploi
FILEMAKER	SGBD
FILMS ACTION	Images de synthèse
FIRST INTERACTIVE	Livres
FORTE SOFTWARE	Intranet
FORUM des Arts de l'Univers Scientifique et Technologique	CD-ROM, site Internet
FREE BSD	Système
GEOWORKS	Système
GNU	Logiciel libre
HAHT SOFTWARE	Intranet
HEWLETT PACKARD	Assistant de poche, carte à puce
IBM	SDK, Intranet
IBM GLOBAL SERVICES	SSII
ICANN	Noms de domaines
ILION	XML
INFORMIX	Intranet, SGBD, Linux
INPI	Propriété industrielle
INPRISE ex BORLAND	SDK, CORBA
INRIA	Informatique et automatique
INSIGNIA	Java embarqué
INSTALLSHIELD	Installation
INTEL	Processeurs, vidéo, 3D
INTER EDITIONS	Livres
INTERSOLV	Outils SDK, gestion de projet
ISE	Eiffel
ISICOM	Fournisseur d'accès Internet
JAVACUP	Concours Java
JAVALOBBY	Logiciels libres Java
JPEG CONSORTIUM	JPEG standard
LE MONDE EN TIC	Livres
LEARNING TREE	Formation
LINUX	Site officiel
LINUX PPC	Système Linux Macintosh
LOTUS	Groupeware, workflow, JavaBean
LUCENT TECHNOLOGIES	CTI
MACROMEDIA	Multimédia, web
MAGIC SOFTWARE	AGL
MANDALA INTERNATIONAL	Linux
Matra Système & Information	SSII
MC2 DIFFUSION	Linux
METROWERKS	SDK
MICRO APPLICATIONS	Livres
MICRO SIGMA	Outils, SDK, carte à puce
MICROSOFT FRANCE	Système, SDK, web
MICROSOFT PRESS	Livres
MIDI CONSORTIUM	Midi Standard
MK LINUX	Système Linux Macintosh
MPEG CONSORTIUM	MPEG Standard
MTIC	Premier ministre
NAT SYSTEMES	Intranet
NETBSD	Système
NETDYNAMICS	Portail
NETOBJECTS	Web
NETQUARTZ	Outil SDK
NETSCAPE	Web, Javascript
NOKIA	Mobiles, PC de poche
NOVELL	Réseau

Société	Spécialités
NUMEGA	Outils SDK
OBJECTDESIGN	XML
ODMG	Base de données objet standard
OMG	Objet, UML et CORBA standard
OMNIS	SGBD
OMPI CONSORTIUM	Propriété intellectuelle internationale
OPEN GL CONSORTIUM	OpenGL standard
OPENBSD	Système
OREGON SCIENTIFIC	Ordinateurs de poche
O'REILLY	Livres
PALMWARE	Applications mobiles et portables
PARASOFT	Outils, SDK
PARC	Recherche Palo Alto
PC SOFT	AGL
PEOPLE SOFT	Workflow
PERL	Perl
PERSISTENCE OF VISION	Pov Ray
PERVASIVE SOFTWARE	Outils, SDK, web
PGP	Chiffrement
PHILIPS	Assistants de poche
PICK SYSTEMS	Linux
PICOBSD	Système
POWER SOFT	Outils, SDK
PROGRAMMER'S PARADISE	Outils, SDK
PROGRESS SOFTWARE	Intranet
PROLOGIN	Concours
PSION	Ordinateurs de poche
PYTHON	Python
QNX	Système
RAINBOW TECHNOLOGIES	Dongles
RATIONAL	UML
REAL SOFTWARE	Real Audio, Real Vidéo
REDHAT	Linux
SCIENTIFIC AMERICAN	Livres
SEAGATE	Reporting
SEKOYA	XML
SOFTEAM	XML
SOFTWARE AG	Outils, web, SGBD XML
SOFTWARE Technology Resources	Outils, SDK
SOS DEVELOPERS	Outils, SDK
STAR DIVISION Corporation	Bureautique, Linux
STERLING SOFTWARE	SDK, outils, UML
SUN MICROSYSTEMS	Java, SDK, système
SYBASE	SGBD, SDK, outils, web
SYBEX	Livres
SYMANTEC	SDK, outils, web
SYNDICAT national des supports d'enregistrement	Statistiques
TAKEFIVE SOFTWARE	Gestion de projet
TELEBIG	SSII
TELXON	PC industriels, middleware
TMO	Sondages
TRON	Logiciels embarqué libre
UIT	Télécommunications standard
VERISIGN	Outils, web
VERSANT	SGBD Objet
VIGNETTE	Outils, Web
VIRTOOLS	Jeux
VRML CONSORTIUM	VRML standard
W3C CONSORTIUM	HTML, XML, DHTML, CSS standard
WYDE INC.	Modeleur

Société	Adresse	Téléphone	Fax	Site Web
ACI	60, rue d'Alsace - 92110 Clichy	01 40 87 92 00	01 40 87 92 01	www.aci.fr
ALADDIN FRANCE	205, avenue Georges Clémenceaux - 92012 Nanterre	01 41 37 70 39	01 41 37 70 39	www.aladdin.fr
ARDENT SOFTWARE Europe	3, place de Saverne 92901 Paris-La-Défense	01 49 04 65 65	01 49 04 65 50	www.ardentsoftware.fr
AWARE	1, rue du Dahomey - 75011 Paris	01 43 56 57 58	01 43 56 66 77	www.aware.fr
COMEDIA	4, bis allée Charles V 94300 Vincennes	01 43 28 80 29	01 43 28 93 80	non disponible
COMPUWARE France	1, avenue de la Cristallerie 92316 Sèvres	01 41 14 20 00	01 46 23 88 89	www.compuware.fr
DASAR	21, rue Pasteur 92300 Levallois-Perret	01 41 27 21 55	01 47 37 52 77	www.dasar.fr
EDITIONS EYROLLES	61, bld Saint-Germain 75240 Paris cedex 05	01 44 41 11 11	01 44 41 46 00	www.eyrolles.com
IBM	Tour Descartes - La Défense 5 92066 Paris cedex	01 49 05 50 00	01 47 89 91 95	www.ibm.fr
INTERNET FR	2-12, chemin des Femmes Immeuble Odysée - 91300 Massy	01 64 53 12 12	01 64 53 12 19	www.internt.fr.net
INTERSOLV	1, place de la Pyramide 92911 Paris-La-Défense	01 41 99 21 00	01 41 99 21 49	www.intersolv.com/france
ISICOM	5-7, rue de l'Amiral Courbet 94160 Saint-Mandé	01 43 98 22 22	01 43 98 88 44	www.isicom.fr
LOTUS	Immeuble Lotus - Défense 6 35/41, rue du Capitaine Guynemer 92925 Paris-La-Défense	01 41 99 58 00	01 41 99 59 99	www.lotus.fr
MCD2 DIFFUSION	5-7, rue Leredde - 75013 Paris	01 56 29 23 00	01 56 29 23 01	www.mcd2diff.fr
MICROSOFT	18, avenue du Québec 91957 Les Ulis Cedex	01 69 86 46 46	01 69 86 47 61	www.microsoft.fr
O'REILLY	18, rue Séguier 75006 Paris	01 40 51 52 30	01 40 51 52 31	www.editions-oreilly.fr
PC SOFT	216, rue des Escarceliers - BP 3019 34034 Montpellier cedex 01	04 67 03 20 32	04 67 03 07 87	www.pcsoft.fr
PROGRAMMER'S PARADISE	Energy 6 150-154, boulevard de Verdun - 92413 Courbevoie	01 49 04 11 12	01 49 04 11 10	www.iway/fr/ppfrance
RAINBOW TECHNOLOGIES	122, avenue Charles de Gaulle 92522 Neuilly Sur Seine cedex	01 41 43 29 00	01 46 24 76 91	www.rainbow.fr
CAMPUSPRESS France et ex SIMON & SCHUSTER Mc MILLAN	19, rue Michel le Comte 75003 Paris	01 44 54 51 10	01 48 04 53 61	www.campuspress.fr
SOFT CONCEPT	71, rue Albert Thomas 69447 Lyon cedex 03	04 78 54 64 00	04 72 33 19 91	www.softconcept.com
STAR DIVISION Corporation	6515 Dumbarton CA94555 Fremont - USA	00 1 510 505 14 78	00 1 510 505 14 75	www.stardivision.com
SUN MICROSYSTEMS et SUN SOFT	13, avenue Morane Saulnier - BP 53 78142 Vélizy cedex	01 30 67 50 00	01 30 67 53 00	www.sun.fr
Software Technology Resources	7, rue Paul Dautier 78140 Vélizy	01 30 70 61 61	01 39 46 38 64	www.str.fr
SOS DEVELOPERS	2, rue des Colannes - Paris Bourse BP 945 - 75002 Paris	01 42 44 11 23	01 42 44 11 31	www.sosdevelopers.com
SYMANTEC	31/35, rue Gambetta 92150 Suresnes	01 41 38 57 00	01 41 38 57 59	www.symantec.fr
TAKEFIVE SOFTWARE	13, avenue Morane Saulnier Le Chavez - 78143 Vélizy	01 34 58 26 35	01 34 58 26 39	www.takefive.com/france