

# Developpez

## Magazine

Edition de Juillet 2005.

Numéro 2.

Magazine en ligne gratuit.

Copies conformes à l'original autorisées.

Directeur de la publication : Bobby Joe Lambert

Réalisation : Hugo Etiévant

Rédaction : la rédaction de Developpez

Contact : [magazine@redaction-developpez.com](mailto:magazine@redaction-developpez.com)

*Developpez.com*

### Java

Les classes de l'API  
Jakarta Commons  
page 2

### Perl

Débutez avec  
facilité et rapidité  
page 10

**Dictionnaire**.....page 6

**HTML**.....page 7

**Delphi**.....page 8

**Perl**.....page 9

**DotNET**.....page 11

**PHP**.....page 13

**Linux**.....page 13

**Liens**.....page 14

### Cours Méthodes

## Extrem

## Programming

*Extrem Programming propose de remplacer certaines notions acquises par des idées révolutionnaires, et de rendre le développement de logiciel efficace et agile*

par **Eric Lagacé**

page 4

### Évènements

## Microsoft

## TechEd 2005

*Les Teched 2005 ont eu lieu à Amsterdam, du 04 juillet*

*2005 au 08 juillet 2005. Voici un compte rendu de cette semaine, dédiée à des conférences sur les produits Microsoft.*

par **LEBRUN**

**Thomas**

page 11



Developpez Magazine est une publication  
du site [Developpez.com](http://Developpez.com)

# Java

## Les derniers tutoriels

### Journalisation avec l'API Jakarta Commons Logging

Nous étudierons ici l'API Commons Logging qui est une couche d'abstraction pour la plupart des systèmes de journalisation existants.

Lire le cours de **Sébastien Le Ray (lien 1)**

### Utilisation de l'API Jakarta Commons Collections

Après une présentation des concepts clés et des entités supportant le fonctionnement

de l'API, nous décrivons, package par package, les fonctionnalités offertes pour chaque type de collection.

Lire le cours de **Sébastien Le Ray (lien 2)**

### Les Annotations de Java 5.0 Tiger

Les Annotations permettent de marquer différents éléments du langage Java avec des attributs particuliers, dans le but d'automatiser certains traitements et même d'ajouter des traitements avant la

compilation grâce au nouvel outil du JDK : apt...

Lire le cours de **F. Martini (lien 3)**

### Java pour le temps réel

Article consacré à l'analyse des différentes techniques permettant de réduire l'écart entre les langages C/C++ et Java dans les systèmes dits en temps réel/embarqués.

Lire le cours de **Jean-Marie Dautelle (lien 19)**

## F.A.Q Général Java

Déjà 280 réponses à vos questions dont 5 nouvelles. Cette faq a été réalisée à partir des questions fréquemment posées sur les forums de Developpez

Lire la F.A.Q : <http://java.developpez.com/faq/java/>

## Outil en ligne : JavaSearch

Ce moteur de recherche dans la JavaDoc J2SE 1.4.2 vous permet de retrouver en un clin d'oeil une API, une méthode, une classe ou même un attribut. JavaSearch est un outil développé et entretenu par Laurent PERRON.

Accès : <http://javasearch.developpez.com>

## Principaux sites Java

-  **JavaSearch** : <http://javasearch.developpez.com/>
-  **Sun Java** : <http://java.sun.com/>
-  **Java.com** : <http://www.java.com/>
-  **Java One** : <http://java.sun.com/javaone/>
-  **Eclipse** : <http://www.eclipse.org/>
-  **ObjectWeb (JOnAS...)** : <http://www.objectweb.org/>
-  **JBoss** : <http://www.jboss.com/>
-  **Bea (Weblogic)** : <http://fr.bea.com/>
-  **Borland (BES, JBuilder)** : <http://www.borland.fr/>
-  **NetBeans** : <http://www.netbeans.org/>

## Notre partenaire

Le **Club-Java** a pour vocation de promouvoir la technologie Java et de la mettre en avant sous forme d'actions de pédagogie, d'aide et d'assistance pour ses adhérents.

Accès : <http://www.club-java.com/>



## Vu dans le blog

### Les nouveautés de Java SE 6 (Mustang)

Voici un petit récapitulatif de ce qu'il y a de neuf dans Java SE 6 (Mustang) :

- Support format PNG pour les icônes
- Amélioration performance fichiers Zip situés sur le réseau
- Possibilité de connaître l'espace disque disponible
- AntiAliasing & LCD AntiAliasing support
- Ajout tri sur JTable
- L'intégration du moteur de javascript rhino dans le JSE.
- La fin du rectangle gris quand on passe une fenêtre au dessus de l'autre en swing
- Un java web start relooké
- Un binding OpenGL pour swing activé par défaut.
- Amélioration fidélité Look and Feel GTK
- Exécution directe des JAR sous Unix

Cela n'est évidemment pas une liste complète. Mais juste quelques points qui me paraissent intéressants d'être mentionnés. Surtout les deux derniers qui devraient grandement intéresser ceux qui utilisent Linux/Unix.

Lire le blog de **vbrabant (lien 4)**

## Vu dans la F.A.Q Java

### Comment signer un JAR ?

Il faut d'abord générer un certificat :

```
keytool -genkey -alias signature -keystore monStore
```

Il faut alors répondre aux différentes questions (nom, prénom, société, adresse,...). Cela génère un certificat "signature" qui est stocké dans un dépôt "monStore"

Ensuite pour signer le jar :

```
jarsigner -keystore monStore -signedjar sMonJar.jar monJar.jar signature
```

Cela génère le jar "sMonJar.jar" qui est la version signée avec le certificat "signature" du jar "monJar.jar".

Lire la Q/R de **christopheJ (lien 5)**

# UML

## Les derniers tutoriels

### L'Extreme Programming

Extreme Programming propose de remplacer certaines notions acquises par des idées révolutionnaires, et de rendre le développement de logiciel efficace et agile.

Lire le cours de **Eric Lagacé** ([lien 6](#))

### eXtreme Programming

Lire le support de cours de **Stéphane Frénot** ([lien 18](#))

### L'Extreme Programming (XP)

L'Extreme Programming (XP) est un processus de développement logiciel, c'est-à-dire un ensemble de pratiques destinées à organiser le travail d'une équipe de développement. Ces pratiques se focalisent sur la construction proprement dite du logiciel, en aval des phases préparatoires d'études d'opportunité ou de faisabilité.

Lire le cours de **Régis Medina** ([lien 7](#))

## Critique de livre

### MDA en action - Ingénierie logicielle guidée par les modèles

Xavier Blanc  
Editions Eyrolles

37,05 EUR



Livraison gratuite sur Amazon



Lorsque que MDA (Model Driven Architecture) est apparu il y a quelques années, bon nombre de développeurs l'a considéré comme un acronyme dissimulant un niveau d'abstraction supplémentaire à la modélisation UML et apportant une complexité peu implémentable dans l'industrie logicielle. Mais voilà, UML a évolué, les enjeux de la modélisation, les outils de conception en entreprise ont changé et l'approche MDA peut-être exploitée, même si la spécification reste floue sur une méthode claire et formelle d'utilisation.

Après la lecture du livre MDA en action, on comprend bien que, sans appliquer tous les concepts avec exactitude, on peut mettre en place une approche de pilotage des applications basée sur les modèles. Il en résulte une définition plus formelle de l'application, un processus de conception maîtrisé et (c'est ce qui intéresse les décideurs), davantage de productivité.

Xavier Blanc est Maître de conférence à l'Université Pierre et Marie Curie. Il enseigne les techniques de modélisation, et représente également le LIP6 à l'OMG. Sa contribution est notamment mesurable au sein

du projet "IST ModelWare", destiné à améliorer l'interopérabilité des outils de modélisation.

L'auteur présente tout d'abord l'architecture MDA et les concepts qui y sont inhérents (modèles CIM, PIM, PSM, MOF, UML, XML). Ce premier chapitre est essentiel à la compréhension des autres chapitres qui amènent plus de profondeur au sujet.

Le chapitre 2 "Modèles et niveaux méta" aborde les points les plus abstraits et les fondements de MDA. On apprécie d'ailleurs, bien que l'effort de lecture soit important, une certaine concision et recherche de la justesse des définitions, illustrés par des exemples. Ce chapitre étudiera donc les notions de modèle, méta-modèle, métaméta-modèle. Cette représentation des modèles est ensuite enrichie au chapitre 3 avec l'étude des méta-modèles d'UML2.0 (UML2.0 Superstructure et UML2.0 Infrastructure, profils UML), qui conjuguent la modélisation de la notation UML et de MOF.

OCL et AS sont ensuite présentés succinctement. J'aurais aimé un dossier plus complet sur le sujet, peut-être dans une version future du livre. L'interopérabilité et la manipulation des modèles sont abordés aux chapitres 5 et 6, où l'on étudiera le standard XMI et les frameworks JMI et EMF.

Un exemple de transformation des modèles est donné à titre d'exemple ; même remarque, c'est concrètement l'un des éléments qui amène davantage de productivité à MDA en entreprise, et j'aurai apprécié une étude de cas professionnelle.

Bien entendu, l'un des grands intérêts de l'ouvrage est de consacrer 2 chapitres à MDA avec la plateforme J2EE et MDA avec la plate-forme PHP, illustrés d'exemples de mise en oeuvre, d'utilisation des Profils UML pour EJB et des transformations de PIM en PSM. Une étude de cas, certes légère puisque il s'agit du célèbre "PetStore" clos l'ouvrage.

Il s'agit donc d'un bon livre, relativement théorique, qui pourra intéresser les architectes et concepteurs d'applications, les chefs de projets et décideurs en quête d'informations sur l'élaboration de processus de développements.

L'ouvrage est packagé avec un CD-ROM contenant les outils pour MDA IBM Rational Software Modeler et Softeam MDA Modeler.

**Pierre Chauvin**

Retrouvez la critique complète ([lien 8](#))

Retrouvez la rubrique UML sur <http://uml.developpez.com>

# UML & Méthodes

## Extrem Programming

*Extrem Programming propose de remplacer certaines notions acquises par des idées révolutionnaires, et de rendre le développement de logiciel efficace et agile.* **Eric Lagacé**

### I. Introduction

Depuis les débuts de l'informatique commerciale dans les années '60, plusieurs méthodologies de développement de logiciel ont vu le jour. Le modèle en cascade et ses dérivés ont connu un grand succès, mais leur lourdeur et rigidité sont de sérieux handicaps. Extreme Programming propose de remplacer certaines notions acquises par des idées révolutionnaires, et de rendre le développement de logiciel efficace et agile.

#### I-A. La petite histoire d'une grande industrie

L'histoire du développement de logiciels est intimement liée à l'histoire de l'industrie informatique. Je vais donc présenter les deux à la fois.

#### I-B. La vieille époque

Le transistor a été inventé en 1948, et l'ordinateur sous sa forme actuelle existe depuis les années '50. Ce n'est cependant que pendant les années '60 que l'informatique commerciale est apparue, alors que le coût des ordinateurs a grandement diminué et que des langages de programmation comme Fortran et Cobol ont permis de produire des applications pour le milieu des affaires. À partir de ce moment, le rythme n'a cessé d'augmenter, tant au point de vue machine que logiciel.

Les années '70 ont vu l'apparition des langages C, Pascal et Basic, de Unix, du circuit intégré, de Apple, Intel et Microsoft, et des premiers micro-ordinateurs. Le PC, le Mac et l'interface graphique ont suivi quelques années plus tard, et une révolution importante s'est produite: un individu pouvait désormais programmer avec un langage comme dBase ou Basic, sans aucune formation et avec un minimum de ressources, et produire un logiciel utile et immédiatement disponible. La qualité était souvent mauvaise, et les règles établies depuis les années '60 ont été oubliées au profit de cette nouvelle vague d'outils de développement. L'avènement

en masse de la réseautique a amplifié le problème, puisque les défauts d'un logiciel n'étaient plus isolés au seul ordinateur où il était exécuté.

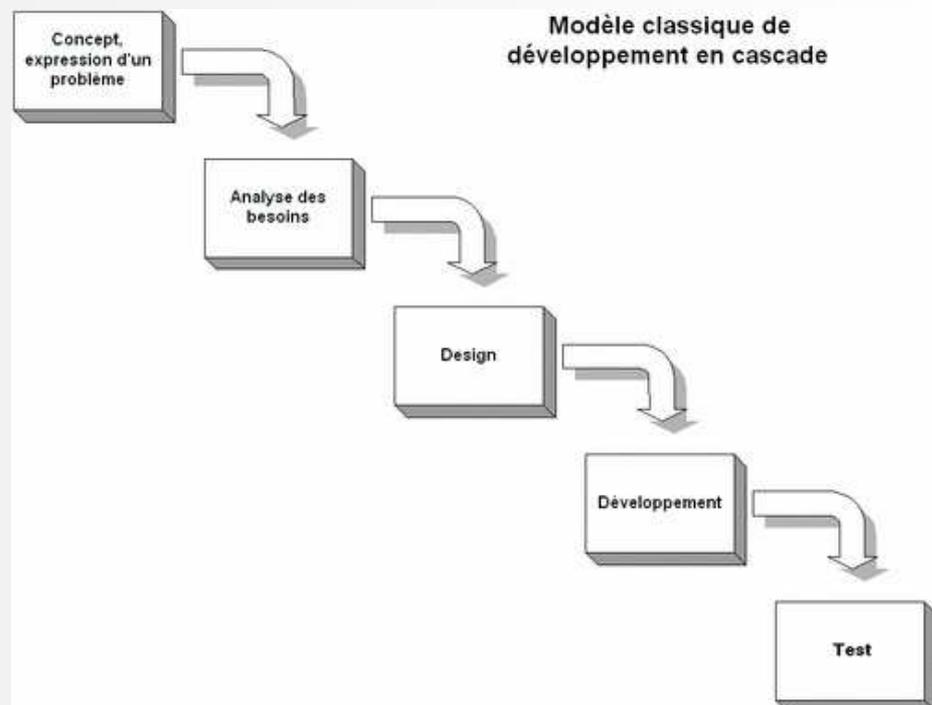
Le développement de logiciels a été pendant longtemps entièrement libre de directives ou de contraintes, et il n'y avait aucune méthode pour analyser les besoins, produire une solution, ou même évaluer le succès. En 1968, une conférence de l'OTAN a introduit le terme « génie logiciel » et suggéré d'utiliser les méthodes rigoureuses et éprouvées du génie civil au chaos du développement de logiciels. L'idée était louable, mais une différence majeure existe: contrairement à la physique et aux mathématiques, l'informatique ne repose sur aucune loi et ne peut être vérifiée scientifiquement.

#### I-C. De la rigueur s.v.p.

Des méthodologies de développement sont apparues à différents moments durant la révolution informatique. Le modèle en cascade, inventé par la US Navy, est sans aucun doute le modèle qui a eu le plus d'influence, et cette influence peut encore être ressentie aujourd'hui. Le modèle est très strict: les étapes de concept, analyse, design, programmation et test doivent être

exécutées dans l'ordre, et le retour en arrière n'est pas permis. La notion que le coût du changement augmente à mesure que le projet progresse est dérivée de ce modèle. L'emphase sur la documentation est très importante, et chaque étape doit être approuvée avant que l'étape suivante débute. Très bureaucratique, très lourd, mais un grand bond dans la bonne direction.

Le modèle en cascade a donné naissance à de nombreuses autres méthodologies: prototypage, modèle en spirale, implantation en étape, et même RAD (Rapid Application Development). Ces méthodologies sont des variations de l'original, où des possibilités de retour-arrière à différents points ont été ajoutées. Au milieu des années '90, l'industrie informatique utilisait le terme RAD à toutes les sauces, et promettait des résultats fantastiques. Les fabricants de langages de programmation, dont Borland et Microsoft, ont lancé des produits de développement qui ont effectivement accéléré le processus de programmation, mais n'avaient rien à voir avec une méthodologie de travail. En fait, dans la grande majorité des cas, les outils contribuaient à diminuer la qualité.



Eric Lagacé est blogueur sur le site *The Maelström*.  
<http://themaelstrom.net>

Ce document est tiré d'articles publiés fin 2003 par Eric Lagacé sur le site *The Maelström*.

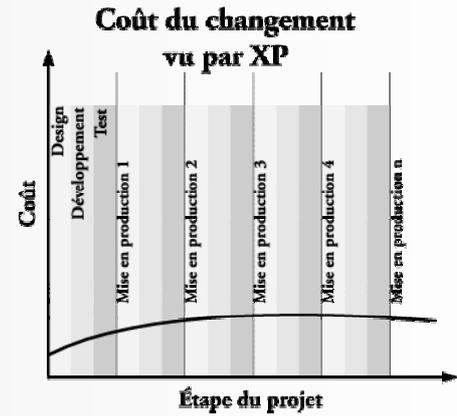
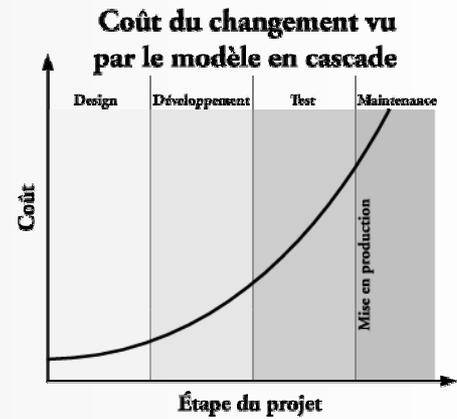


## The Maelström

La majorité des collaborateurs sur ce blog sont québécois ; ils oeuvrent tous dans le domaine des technologies de l'information. Ils sont architecte, administrateur, designer ou programmeur et ce sont tous des passionnés de leur art.

Ce blog collaboratif est un vortex composé de leurs intérêts communs, de leurs idées disparates et de leurs points de rencontre.

D'autres modèles sont apparus et ont connu un succès limité, possiblement à cause de leur manque de flexibilité. L'analyse orientée-objet (basée sur la programmation orientée-objet) est apparue à la fin des années '80, et est encore utilisée aujourd'hui pour le développement de systèmes de grande taille. Rational Unified Process (processus basé sur UML) est le modèle de Rational Software, et requiert l'utilisation de logiciels spécifiques. La méthode P-Plus, de DMR (maintenant Fujitsu Consulting), qui a connue son heure de gloire dans les années '80, commençait par la fin: le livre d'instructions du produit fini était écrit, puis présenté au client pour approbation; le développement du logiciel commençait ensuite.



### I-D. Être agile

À travers toutes ces méthodes et tous ces modèles, de nombreuses bonnes idées sont apparues. Malheureusement, elles sont souvent mal utilisées, ou perdues au coeur de la rigueur et de la paperasse générée par le modèle. Un mouvement est apparu au milieu des années '90 qui avait pour but de rendre le processus de développement plus efficace, en utilisant des concepts très simples et reconnus. Ce mouvement s'appelle Agile Modeling. Il n'est directement relié à aucune méthodologie, mais propose une série de recommandations, valeurs, et principes qui sont appliquées dans d'autres méthodes. C'est de là que Extreme Programming est né.

Extreme Programming, ou XP, est basé sur des principes très simples, mais souvent ignorés par l'industrie. Une des idées révolutionnaires est que le coût du changement n'est pas variable mais plutôt constant; XP accepte donc le changement comme une réalité et l'intègre dans le processus de développement. Aussi, la programmation en paire, où deux programmeurs travaillent ensemble sur le même ordinateur, permet d'augmenter la qualité à des niveaux encore jamais vus. Une autre idée consiste à mettre l'emphase sur la communication constante entre l'équipe de développement et le client, par le biais de boucles rapides développement-test-feedback. Certaines de ces idées ne sont pas nouvelles, mais elles sont assemblées dans XP pour former une méthodologie efficace et qui présente des résultats concrets, plutôt que des résultats sur papier.

## II. Gestion du changement

Depuis les débuts de l'informatique commerciale, le changement a été considéré comme un indésirable ayant pour effet d'augmenter le coût d'un projet. Partant du principe que le changement est inévitable, Extreme Programming intègre les demandes de changement dans le processus de développement, et prétend même pouvoir

contrôler le coût par le biais de mises en productions fréquentes.

XP traite le changement comme une réalité plutôt qu'un indésirable, et propose une approche différente pour l'intégrer au processus de développement.

### II-A. Le passé

Les cours de génie logiciel, la littérature, et presque toutes les méthodologies de développement jusqu'au milieu des années '90 suggéraient que le coût relié à des changements dans les spécifications d'un problème ou le design d'une solution augmente de façon exponentielle à mesure que le projet avance. D'innombrables références et études sont disponibles dans des livres et articles, et l'idée est une notion acquise pour plusieurs générations de gestionnaires de projet. En fait, certaines méthodologies basées sur le modèle en cascade reposent entièrement sur cette prémisse et ignorent complètement les requêtes de changement, ou encore imposent des processus très stricts pour gérer ces requêtes (possiblement dans le but de les éliminer).

Dans la réalité, quelqu'un a-t-il déjà travaillé sur un projet d'informatique (ou n'importe quel projet) où il n'y a pas eu de changement? Est-ce même possible? Peu importe la taille du projet, peu importe le contexte, peu importe l'industrie, le changement est inévitable. Depuis qu'il y a des projets d'informatique, il y a des requêtes de changement; pourquoi les

ignorer? Pourquoi supposer que ce projet sera différent?

### II-B. Retour vers la réalité

Le court cycle de développement utilisé dans XP règle le rythme du projet, et a un effet direct sur la qualité du produit fini. En ayant des mises en production fréquentes, les éléments les plus importants sont produits en premier; les mises en production suivantes permettent d'ajouter des fonctionnalités tout en testant le résultat à chaque nouvelle étape.

Le graphique présente le coût du changement d'après XP. On peut voir que le coût du changement augmente légèrement durant les premières étapes, alors que le design devient plus complexe et que l'effet d'un changement a de plus en

plus de répercussions. Après un certain point, cependant, le coût atteint un plateau et diminue légèrement; ceci est causé par la « factorisation » du programme (expression pompeuse pour dire « simplification »).

Une des grandes forces de XP est d'accepter le changement comme une réalité, et de l'inclure comme partie intégrale du processus de développement. Les boucles rapides de développement-test-feedback permettent d'intégrer le changement de façon contrôlée, sans avoir à "revenir en arrière". En permettant au changement de prendre place à tout moment, il est possible de prédire le temps et le coût associé, et de rapidement présenter au client la conséquence de ce changement. Le coût du changement devient donc une constante plutôt qu'une variable.

Bien entendu, une requête qui aurait des conséquences profondes sur le design d'une base de données ou d'une architecture devra être évaluée sérieusement, et le client sera informé rapidement du coût en temps et ressources; il est possible qu'une requête de changement soit rejetée. De la même manière, il serait naïf de complètement ignorer les réalités économiques du projet, et d'accepter d'implanter des changements de toutes sortes, n'importe quand, tout en continuant de viser une date de livraison fixe. Le gestionnaire du projet garde le doigt sur le pouls des variables: temps, étendue (scope), ressources, qualité; son intervention permet de garder le projet à un niveau réaliste tout en fournissant au client un maximum de résultats pour son investissement.

Découvrez la suite de ce cours sur Developpez : <http://lagace.developpez.com/extreme-programming/>

# Dictionnaire

## Le mot du mois

Sécurité > IDS

### IDS

**Intrusion Detection System**  
sigl.

Terme générique faisant référence aux équipements ou logiciels chargés de détecter des intrusions. Les IDS permettent de garder une trace d'événements anormaux, de signaler en temps réel des opérations jugées illégales et même de réagir sur la base de signatures d'attaques ou l'analyse de comportements (heuristique). On distingue deux types d'IDS : Les NIDS pour les réseaux et les HIDS pour les serveurs.

IDS  
⇕  
NIDS, HIDS

IPS (SI)

Source : Alain Ceccato - ICBO SARL

Permalien : <http://dico.developpez.com/html/931-Securite-IDS-Intrusion-Detection-System.php>

Date d'ajout : 18/03/2005      Date de dernière mise à jour : 18/03/2005

**1225 mots**  
**44 contributeurs**  
**10 thèmes**  
**100 ressources**  
**90 illustrations**



Fort de près de 1200 mots et de 44 contributeurs, le Dictionnaire des développeurs francophones croît un peu plus chaque jour.

Avec une moyenne de 1000 définitions vues par jours, cet outil séduit de nombreux développeurs qui souhaitent élargir leur connaissance du jargon informatique.

De nombreuses thématiques y sont représentées : la culture, l'économie, l'infographie, le web, la programmation, la gestion de projet, la sécurité informatique, le matériel et les télécoms.

Retrouvez le dico sur : <http://dico.developpez.com>

# HTML

## La structure des balises : bloc et en-ligne

La compréhension de la structure des éléments va avoir de nombreuses implications par la suite, et paradoxalement c'est souvent un sujet peu connu des webmasters. **Raphaël Goetter**

Les balises HTML ont une structure particulière. En fait, il existe deux grands groupes principaux de balises : les balises de type "bloc" et les balises de type "en-ligne" (ou "inline").

Ceci est particulièrement important car de la structure des balises vont dépendre tous leurs comportements, positionnements, affichages, imbrications, etc.

On peut comprendre assez facilement la différence fondamentale entre ces deux structures d'éléments : les éléments "bloc" servent à distinguer les parties entières de texte, comme des titres, des paragraphes, des listes, des citations, etc. ; les éléments "en-ligne" sont prévus pour rester dans le texte pour l'enrichir (lien hypertexte, emphase, renforcement, etc.).

Exemples de balises de type en-ligne : SPAN, EM, STRONG, IMG, BR, INPUT, etc.

Exemples de balises de type bloc : DIV, P, H1...H6, UL, OL, TABLE, PRE, etc.

### Conséquence sur le positionnement

De ces deux groupes découlent des spécificités d'affichage :

- les "blocs" se placent toujours l'un en dessous de l'autre par défaut (comme un retour chariot). Par exemple: une suite de paragraphes (balise <p>) ou les éléments d'une liste (balise <li>).
- les "en-ligne" se placent toujours l'un à côté de l'autre afin de rester dans le texte. Par exemple : le renforcement d'une partie de texte à l'aide de la balise <strong>.

Par exemple :

```
<p>Paragraphe 1</p><p>Paragraphe 2</p>
```

Ces deux paragraphes vont s'afficher sur deux lignes, car la balise <p> est une balise de type "bloc" : chaque paragraphe va occuper une ligne.

Autre exemple :

```
<strong>Toto</strong> et <em>moi </em>
```

Ce texte va s'afficher sur une seule ligne (aucun retour à la ligne) car les balises qui le définissent sont de type "en-ligne".

### Imbrications et dimensions

Au niveau structurel et imbrications, il y a également des différences notables :

- Une balise bloc peut contenir une (ou plusieurs) balise bloc et/ou en-ligne, sauf exceptions, et avoir une dimension (largeur, hauteur définies).
- Une balise en-ligne ne peut contenir QUE une (ou plusieurs autres) balise en ligne.

Les éléments en-ligne se distinguent eux-même en deux parties : les éléments "remplacés" et les éléments "non-remplacés".

Les éléments "remplacés" sont les seuls qui possèdent des dimensions (height, width) par défaut. Il s'agit des éléments <img>, <input>, <textarea>, <select> et <object>.

Tous les autres éléments en-ligne ("non-remplacés") n'ont pas de dimension à proprement parler (il n'occupent que la place minimum nécessaire à leur contenu). C'est le cas des éléments <strong>, <em>, etc.

Par exemple :

```
<div><p>Paragraphe 1</p><p>Paragraphe 2</p></div>
```

La balise <div> (bloc) englobe les deux paragraphes (blocs). Ceci est autorisé.

Par contre, nous n'aurions pas pu écrire :

```
<span><p>Paragraphe 1</p><p>Paragraphe 2</p></span>
```

car la balise <span> (balise en-ligne) n'est pas autorisée à contenir des éléments blocs comme les paragraphes.

Une mise en page se fera donc à l'aide de balises blocs, la balise <div> étant la plus indiquée pour cet usage car elle est une balise générique servant de conteneur, de zone neutre (elle signifie Diviseur).

### Les balises en-ligne

Les balises, ou éléments, "en-ligne" sont également appelées "balises internes" car leur but est de donner du sens à des parties de texte ou de bloc, tout en restant dans le texte. C'est le cas par exemple des éléments de renforcement (gras pour les

navigateurs graphiques), et les éléments d'emphase (italique).

Les balises en-ligne sont faites pour rester au sein du texte pour l'enrichir et lui apporter du sens. Il n'est pas prévu qu'elles puissent se positionner sur la page (même si cela est possible), ni de leur donner des dimensions (hauteur, largeur, profondeur) : exception faite des balises en-ligne "remplacées" (cf explications plus haut), leur taille va être déterminée par le texte ou l'élément qu'elles contiennent.

Par défaut, les balises en-ligne ont des marges internes et externes inexistantes (valeur nulle), contrairement aux balises bloc.

### Les balises bloc

Les balises de "bloc" ont une structure qui leur permet de bénéficier de dimensions (hauteur, largeur, profondeur), de contenir d'autres éléments dimensionnés, de posséder des marges internes (padding) et externes (margin) mais également, et surtout, d'être positionnés, c'est à dire de contribuer à la mise en page du document. La plupart de ces propriétés sont réservées aux éléments bloc. Par exemple, comme nous l'avons expliqué, une balise en-ligne comme <em> ne pourra pas posséder de dimensions propres, celles-ci dépendront du contenu (texte) de la balise.

Notez que vous pouvez aisément passer d'une structure bloc à une structure en-ligne (et vice-versa) grâce à la propriété CSS "display".

Par défaut, la plupart des éléments bloc (en fait, tous sauf la balise neutre <div>) possèdent des marges internes et externes non nulles. Ce détail est important car ces marges sont souvent différentes selon les navigateurs et il faut parfois les annuler pour éviter de grosses différences de rendu visuel. La prise en compte de ces marges par défaut peut vous permettre d'éviter de gros soucis de compatibilité entre les navigateurs.

Retrouvez cet article en intégralité en ligne ([lien 9](#))

Raphael Goetter est webdesigner et intégrateur XHTML / CSS. Il est l'auteur du livre **CSS 2 - Pratique du design web** paru aux éditions Eyrolles.



# Delphi

## Les derniers tutoriels

### Création de composants

*Vous allez apprendre à réaliser des composants non visuels, c'est-à-dire des composants qui apparaissent sur votre fiche uniquement en conception (comme le TTimer par exemple)*

Lire le cours de **Sébastien Doeraene** ([lien 10](#))

### Les énumérations sous Delphi .NET

*Les énumérations du framework .NET me semblaient être un type de donnée différent mais il n'en est rien, elles se manipulent de la même manière qu'une énumération déclarée sous Delphi .NET. Vous trouverez dans ce tutoriel les manipulations de base sur les énumérations sous Delphi .NET ainsi*

*qu'une approche de la technique d'injection de code CIL au travers de la classe EnumBuilder.*

Lire le cours de **Laurent Dardenne** ([lien 11](#))

### La gestion des exceptions et le contrôle d'erreurs

*Cet article vous donne tout ce que vous avez besoin de savoir pour vous sentir à l'aise avec la bête*

Lire le cours de **Tony Baheux** ([lien 12](#))

### Présentation et utilisation des menus avec Delphi

Lire le cours de **Jean-Michel Doudoux** ([lien 13](#))

### La création d'objets COM pour ASP

Lire le cours de **Jean-Michel Doudoux** ([lien 14](#))

### Présentation et utilisation des fiches avec Delphi

Lire le cours de **Jean-Michel Doudoux** ([lien 15](#))

### Utiliser une BDD Firebird sous Delphi 7 / 2005 Personnel

*Ce cours présente comment attaquer une base de données Firebird sous Delphi 7/2005 édition personnelle (win32). En utilisant la bibliothèque PythonForDelphi*  
Lire le cours de **Guillaume Duriand** ([lien 20](#))

## Les meilleurs outils

Les membres des forums Delphi de Developpez ont voté pour les produits Delphi qu'ils préfèrent.

- Meilleur composant ou ensemble de composants Delphi gratuit : **Jedi VCL (JVCL)**
- Meilleur outil, plug-in ou expert Delphi gratuit : **GExperts**
- Meilleur composant ou ensemble de composants Delphi commercial : **TMS Component Pack Pro**

Voir les résultats détaillés des votes : <http://delphi.developpez.com/freewares/elections2004/>



## Critique de livre

### Delphi 8 pour .NET

Olivier Dahan

738 pages, Juillet 2004, Editions Eyrolles, ISBN: 2212113099, 42,75 EUR

Encore un ouvrage de référence à posséder ! Ce livre est le complément idéal de la documentation livrée avec Delphi 8.

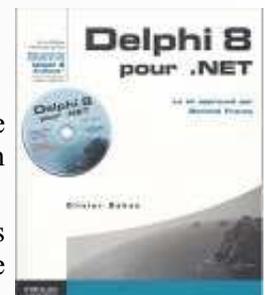
Il présente de façon clair les nouveautés apportées par le Framework. Ce que cela change pour les habitués de Win32, ce qui ne change pas et nous oriente tout de suite vers les bonnes habitudes à prendre avec Delphi pour .NET

Une partie de l'ouvrage est dédiée à la migration des applications et des composants vers Delphi

.NET, c'est sans doute le premier chapitre que l'on lit quand on veut commencer la migration des applications.

Une remarque cependant, ce livre n'est pas destiné aux débutants avec Delphi. Il suppose que le lecteur soit déjà un habitué de Delphi, sans être forcément un expert.

Retrouvez la critique complète ([lien 16](#))



**Nono40**

Livraison gratuite sur Amazon



Retrouvez la rubrique Delphi sur <http://delphi.developpez.com>

# Perl

**Nouvelle rubrique !**

## Les cours

### Débutez en Perl (101 pages, 190 Ko)

Cours d'introduction pour les débutants.

Cours de **François Dagorn** et **Olivier Salaun**

### Documentation Perl en Français (513 pages, 1.4 Mo)

Ce document est la traduction de la documentation originale distribuée avec perl version 5.00502

Cours de **Paul Gaborit** et al.

Tous les cours : <http://perl.developpez.com/cours/>

## Le forum

Le forum Perl de Developpez est très actif, rejoignez nous ! ([lien 17](#))

## Vus dans la F.A.Q

### Comment définir et utiliser une table associative ?

Une table associative est une variable dont le sigill (le caractère qui précède le nom) est % :

```
my %association = (1 => 'aaa', 2 => 'bbb', 3 => 'ccc');
```

L'usage de cette table se fait de deux manières, selon que l'on veut la manipuler dans sa globalité, ou bien accéder à un élément particulier. L'accès global au tableau associatif se fait avec la notation en % :

- %hash = (); pour vider un tableau de hachage,
- %hash = (1 => 'premier', 2 => 'deuxieme', 3 => 'troisieme'); pour affecter une liste associative à un tableau du même métal,
- %hash = %liste; pour affecter le contenu d'un tableau associatif à un autre tableau de hachage (attention, il s'agit d'une duplication du contenu),
- %hash = (%hash1, %hash2, %hash3); pour affecter le contenu de trois tableaux associatifs dans un autre. Les clefs présentes dans plusieurs tableaux simultanément ne se retrouveront qu'en un exemplaire dans le hachage résultant avec pour valeur associée la dernière valeur rencontrée,
- %hash = @liste; pour affecter une liste simple à une liste associative. Les éléments de la liste simple sont pris par paires, le premier de chaque paire étant utilisé comme clef, le second comme valeur.

Par contre, l'évaluation d'une table associative en contexte scalaire n'est, à priori, pas une idée pertinente.

**2Eurocents**

### Comment récupérer les paramètres passés à une fonction ?

Lors de l'appel à une fonction, PERL range les paramètres dans une liste spéciale, @\_, interne à la fonction.

Cette liste se comporte comme toutes les autres. Il n'est, par contre, pas recommandé de tenter de la modifier. Il est donc possible de récupérer les valeurs de différentes façons :

```
my ($param1, $param2) = @_;
```

récupère les 2 premiers paramètres (@\_ est inchangée).

```
my ($param1) = @_;
```

est équivalent à `my $param1 = $_[0];`

```
my $param = shift;
```

récupère le premier paramètre de la liste et le supprime d'y-celle.

Le nombre de paramètres effectif étant laissé libre, il peut être pratique de le connaître, ne serait-ce que pour effectuer le nombre de shift correct pour récupérer toutes les valeurs. Le nombre de paramètres reçus se trouve simplement par l'évaluation de @\_ en contexte scalaire (`$i = @_;` ou bien `$i = scalar (@_);`).

**2Eurocents**

Déjà 94 questions/réponses dans la F.A.Q !

<http://perl.developpez.com/faq/>

Retrouvez la rubrique Perl sur <http://perl.developpez.com>

# Perl

## Apprendre rapidement le Perl

L'objet de cette première leçon est de vous initier au langage Perl le plus simplement possible. Bien sûr, nous serons trop vous recommander de lire en complément la FAQ Perl ainsi que les cours de la rubrique. **GLDavid**

Perl est un langage interprété créé par Larry Wall qui cherchait un langage de script plus commode à utiliser que awk sous Unix. C'est pour quoi Perl est un langage particulièrement recommandé pour l'extraction de données, ne serait-ce que par la puissance de ses expressions régulières. Citons l'importance de la communauté Perl, à commencer par la richesse de ses modules sur CPAN (<http://www.cpan.org/>) et de nombreux groupes de développeurs tels les célèbrissimes mongueurs de Perl (<http://www.mongueurs.net/>).

Dans cette première approche, nous écrirons des programmes simples en Perl et apprendrons à utiliser les 3 types primitifs de ce langage. Commençons donc à écrire notre premier programme nommé `hello.pl` qu'on exécute en tapant dans le terminal : `perl hello.pl`

```
#La ligne Shebang pour définir
#le path de l'interpréteur
#!/usr/bin/perl
#On affiche une chaîne de caractère
print "Bonjour tout le monde !";
```

Perl est un langage non fortement typé au regard d'autres langages comme C ou C++. Toutefois, il existe des fonctions permettant de déterminer quel type est contenu dans votre variable. Il existe 3 types de variables: les variables scalaires, les listes et les tables associatives. Une variable scalaire est la variable la plus élémentaire de Perl. Vous pourrez y stocker aussi bien des chaînes de caractères, un caractère ou un nombre. Envisageons ce code simple d'addition:

```
#!/usr/bin/perl
print "Calcul d'un prix TTC\n";
#On initialise la valeur de notre taxe
$taxe = 0.20;
#On fixe un prix hors taxe à 100
$pht = 100;
#D'où la valeur ttc
$ttc = $pht+($pht*$taxe);
print "Le prix TTC est de : $ttc.\n";
```

Ce code simple permet ainsi de calculer un prix codé en dur. Naturellement, nous voulons que notre code interagisse avec l'utilisateur:

```
#!/usr/bin/perl
print "Calcul d'un prix TTC\n";
#On initialise la valeur de notre taxe
$taxe = 0.20;
#On demande de rentrer un prix hors taxe
$pht = <stdin>;
#On enlève le caractère de retour chariot
chomp $pht;
#D'où la valeur ttc
$ttc = $pht+($pht*$taxe);
print "Le prix TTC est de : $ttc.\n";
```

Mais, il est évident que nous n'aurons pas que des simples variables à utiliser, nous aurons sans doute à gérer des collections de variables unitaires. Les listes sont idéales pour ça. En Perl, dans une liste, vous pourrez stocker des données non plus homogènes comme en C mais hétérogènes. D'autre part l'accès à une valeur particulière d'une liste s'effectue aussi simplement qu'un accès dans un tableau en C :

```
#!/usr/bin/perl
```

```
print "Des listes et des boucles !\n";
#Un tableau bien hétérogène !
@tablo = (1, 2, 3, "foo", 4, "bar");
#Et une boucle une !
foreach $valeur (@tablo){
#Au passage, faisons une petite concaténation
    print $valeur."\t";
}
#Accès à une valeur particulière
print "\nEt si je veux la 4ème valeur du tableau
? \n";
print "La 4ème valeur du tableau est $tablo[3]\n";
```

Vous aurez remarqué comment nous aurons initialisé notre liste. La boucle `foreach` nous permet à chaque tour de boucle d'affecter la variable scalaire avec la valeur en cours de la liste `tablo` (noté en Perl `@tablo` pour `@rray`). Enfin, l'accès à une valeur s'effectue de manière naturelle comme dans beaucoup de langage en tenant compte du fait que la première valeur d'une liste est l'élément 0.

Il nous reste à aborder les listes associatives. Elles répondent à un modèle clé=>valeur mais sans ordre particulier en comparaison des listes que nous avons vu précédemment. Dans notre exemple, nous allons utiliser les listes associatives pour stocker des urls :

```
#!/usr/bin/perl
print "Quelques URLs intéressantes\n";
#Les clés à gauche de la grosse flèche,
#les valeurs à droite de la grosse flèche
%urls = (
"faq_perl"=>"http://perl.developpez.com/faq",
"faq_linux"=>"http://linux.developpez.com/faq",
"faq_java"=>"http://java.developpez.com/faq");
#Une boucle bien utile pour les listes
#associatives avec une chouette fonction
print "Listes des urls :\n";
#Une boucle intéressante pour récupérer
#les clés et les valeurs
while (($cle, $valeur) = each(%urls)){
    print $cle." a pour url ".$valeur."\n";
}
#Pour accéder à une valeur particulière si
#on connaît sa clé
print "On ne retient quand même que l'url de la
FAQ Perl qui est ".$urls{"faq_perl"}." :\n";
```

Dans ce code, nos clés sont des chaînes de caractères ainsi que les valeurs. Mais vos clés et valeurs peuvent être aussi bien des nombres. Dans notre boucle, nous récupérons la paire clé-valeur en cours grâce à la fonction `each`. Nous demandons ensuite d'afficher cette paire dans une chaîne concaténée. Enfin, l'accès à un élément en fonction de sa clé est simple mais répond à l'annotation suivante : `$nom_de_mon_hash{ma_clé}`.

J'espère ainsi vous avoir fait touché la simplicité de ce langage qui est Perl. Nombreux nous étions à croire que Perl est complexe mais la pratique révèle que ce langage est aussi simple que les autres et qu'il s'avère particulièrement puissant pour la gestion d'un système, la mise en place d'un système d'information ou même le développement d'IHM avec les modules Tk, Gtk ou WxWindows. N'hésitez pas ainsi à consulter notre FAQ Perl pour approfondir cette petite approche. Enfin, venez nous voir sur le forum Perl ! :D (Retrouvez ce cours [lien21](#))

# DotNET

## Les derniers tutoriels

### SMO : SQL Server 2005 et .Net 2.0

Sql Server Management Objects ou SMO est un ensemble de classes permettant d'accéder à la quasi totalité des fonctions de maintenance et d'administration de SQL Server, et cela, directement depuis votre code. Il est donc désormais possible d'effectuer une multitude d'actions dans SQL Server directement depuis une application .Net .

Lire le cours de **Ronald VASSEUR** ([lien 22](#))

### Personnalisation de MSN 7 avec C#

Ce cours présente le développement d'un programme C# permettant la personnalisation de la nouvelle zone de message personnel de MSN7

Lire le cours de **François Laloux** ([lien 23](#))

### La compression de fichier avec .Net 2.0

Nous allons voir dans ce tutoriel comment il est possible avec le Framework .Net 2.0 de compresser ou de décompresser un fichier au format GZip. Cette nouveauté fait suite aux demandes des développeurs qui souhaitent que le Framework intègre de telles fonctionnalités. Toutes ces opérations sont désormais rendues possibles par l'introduction d'un nouveau namespace : le System.IO.Compression .

Lire le cours de **Ronald VASSEUR** ([lien 24](#))

### Internationalisation d'une application asp.NET

Il existe toute une série de solutions pour internationaliser un site. On peut effectivement créer différentes pages en

différentes langues et, selon le choix de l'utilisateur, rediriger vers ces pages. Une autre solution est d'utiliser une page dont on modifie le texte selon la langue et la région choisie. Ce texte peut provenir d'une base de données, d'un fichier XML ou d'une toute autre source de données.

Lire le cours de **Didier Danse** ([lien 25](#))

### ASP.Net 2.0 : les Master Pages

Dans ce tutoriel nous allons découvrir une grande nouveauté d'ASP.Net 2.0 : les master pages. Notre but ici ne sera pas d'aller dans le détail mais plutôt d'appréhender ce que cela va changer dans notre façon de concevoir nos sites et quels sont les bénéfices que l'on peut en tirer..

Lire le cours de **Ronald VASSEUR** ([lien 26](#))

## Compte Rendu des TechEd 2005

### I. Introduction

Les TechEd 2005 sont un ensemble de conférences sur les différents produits Microsoft (Framework .NET, IIS, etc....). Pour les développeurs (ou les IT Pro), c'est LE séminaire européen à ne pas rater, car il permet de rester au courant des évolutions et des possibilités de ces produits.

J'ai eu la chance d'être invité, par l'équipe Académique de Microsoft France, à cette semaine extraordinaire: j'en profite donc pour les remercier et pour vous faire un compte rendu de ce qu'il s'est passé durant la semaine.

### II. Jour 1: Pré-Conférence

Ce jour-là, nous avons assisté à une pré-conférence, c'est-à-dire que nous pouvions choisir une conférence à voir, en avant première, avant que le reste des participants n'arrive le lendemain pour l'ouverture officielle des TechEd.

Pour ma part, j'ai assisté, avec Patrice Lamarche et Benjamin Gauthey (2 autres MVS), à la session présentée par Jeff Prosize (Wintellect) à la session intitulée: "The Zen of ASP.NET 2.0".

Une session intéressante, bien que quelque peu déjà vue, qui présente les nouveautés d'ASP.NET 2.0. Il est vrai que tous les débuts étaient déjà connus (master page, etc.)

mais la deuxième partie de la session (dans l'après-midi) était un peu plus passionnante car elle abordait de nombreuses nouveautés (pour moi en tout cas :)

Jeff Prosize a vraiment réussi un beau challenge car il est vrai que faire une conférence qui dure toute la journée (8h au total) ne doit pas être évident: il a donc été applaudi comme il se doit à la fin de la présentation.

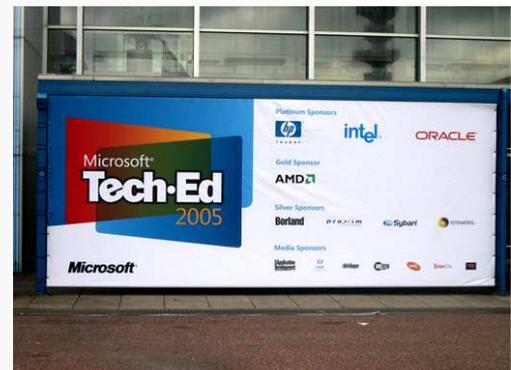
### III. Jour 2: Keynote et Après-Midi Académique

Le deuxième jour, nous avons eu droit à deux parties différentes: le matin, nous avons assisté au Keynote, puis l'après-midi, il a été question pour nous, les étudiants, de passer un moment "entre-nous": c'est ce que l'on a appelé "l'après-midi Académique"

#### III-A. Le Keynote

Le Keynote est une session d'informations qui nous a été présentée par Andy Lees (Corporate Vice President, Server And Tool Business).

L'objectif de cette conférence était simple: "Comment améliorer la productivité de développement d'une application, en utilisant la gamme de produits Microsoft: Windows Server, Visual Studio, etc."



## Developpez Magazine

Pour cela, nous avons eu droit à plusieurs démos, réalisées par différents dirigeants de chez Microsoft Corporation:

- Francois Ajenstat (Group Product Manager SQL Server 2005) qui nous a fait une démonstration des performances de SQL Server 2005, en comparant ce dernier à son prédécesseur, SQL Server 2000
- Brian Keller (Product Manager Visual Studio 2005), qui nous a fait une petite démonstration de Visual Studio 2005
- Josh Cohen (Program Manager), qui nous a montré que WS-Management pouvait aussi bien gérer les systèmes Windows que les systèmes Solaris, etc..
- Mike Hall (Technical Product Manager), qui nous fait une démonstration Windows Mobile, sur comment réaliser la sécurité entre un serveur Microsoft Exchange et un PDA

Lors du Keynote, Andy Lees nous a annoncé, en exclusivité, la date officielle de la sortie de SQL Server 2005-Visual Studio 2005-Biztalk 2006: vous pouvez donc noter, sur vos agendas, cet évènement pour la semaine du 07 Novembre 2005.

### III-B. L'après-midi Académique

L'après-midi Académique était un après-midi, organisé par Microsoft, pour les étudiants, avec des sessions rien que pour nous (dont une session de Rafal Lukawiecki).

La session technique, présentée par Rob Macdonald, était une session de présentation de "WeFly247", le projet .NET développé et livré avec Microsoft Visual Studio 2005 BETA 2, afin de permettre aux développeurs de comprendre les nouveaux concepts et les nouvelles possibilités du Framework .NET 2.0: c'est une solution complète, regroupant des projets divers et variés (ASPX, Web Services, Application

mobiles, Automation Office, etc.), le tout en C# 2.0.

A la fin de la journée, nous avons eu droit à une soirée organisée par Microsoft, et qui se passait à l'"Heineken Experience". Nous avons donc commencé par visiter le musée de l'histoire d'Heineken, puis nous avons terminé la soirée dans les locaux du musée, où nous attendais tout ce que l'on peut rêver pour passer une bonne soirée: musique, jeux vidéos (arcade et réalité

Technologies). Comme son nom l'indique, il s'agissait d'une session sur les VSTO. L'interaction entre Office 2003 et Visual Studio est totale dans cette nouvelle version, ce qui permet un développement grandement simplifié.

Ensuite, j'ai assisté à une excellente session de Brian Noyes (Principal Software Architect) sur "Windows Forms: Making the most of Winforms 2.0 Databinding". Brian nous a rappelé les



virtuelle), et... bière à volonté ;)

### IV. Jour 3: Conférences

Ce troisième jour fût l'occasion d'assister à pas mal de conférences.

Le première session que j'ai vue était celle de Tony Goodhew (Developer Division) et s'intitulait: "Building Smart Client Applications with .NET: The Future of The Software Development".

La deuxième conférence de la journée s'intitulait "Building Office Solutions Using Visual Studio 2005 Tools for Office", et était menée, d'une main de maître, par Ken Getz (MCW

règles du DataBinding et nous a fait une démonstration de cette approche, avec le framework .NET 2.0.

Pour finir ma journée, je me suis autorisé une session moins "code" mais plus portée sur "le test". En effet, j'ai assisté à une conférence intitulée "Visual Studio Team System: Enabling better Software Through Better Testing", et animée par Michael Leworthy (Product Manager). Cette session était donc axée sur Visual Studio Team System et sur les tests unitaires qu'il est possible de réaliser.

[...]

Retrouvez le reste de ce compte-rendu de **LEBRUN Thomas** sur :  
<http://morpheus.developpez.com/TechEd-2005/>

### FAQ Delphi .NET

La réponse à 54 questions couramment posées sur le forum Développement .NET Vous aussi, participez à son élaboration ! ([lien 27](#))



Retrouvez aussi les autres FAQ :

- .NET ([lien 28](#))
- ASP.NET ([lien 29](#))
- C# ([lien 30](#))
- VB.NET ([lien 31](#))

Retrouvez la rubrique DotNET sur <http://dotnet.developpez.com>

# PHP

## Les derniers tutoriels

### Stocker des images dans MySQL

Ce tutoriel va vous apprendre comment stocker depuis PHP, des images (GIF ou JPEG) directement dans une base MySQL. Les images sont directement enregistrées en tant que binaire dans une table et n'ont ainsi aucune présence dans le système de fichiers.

Lire le cours de **Cyril Beaussier** ([lien 32](#))

### Interface XUL pour PHP

Ce tutoriel explique comment réaliser une application basée sur l'architecture client-serveur : comment concevoir une interface utilisateur graphique en XUL, un service Web en PHP et les lier au moyen du protocole SOAP.

Lire le cours de **Pawel Kadluczka** ([lien 33](#))

### Frameworks pour PHP : comment créer efficacement des applications web

Ce cours explique pas à pas comment créer son propre framework PHP..

Lire le cours de **Pawel Kozlowski** ([lien 34](#))

Retrouvez la rubrique PHP sur <http://php.developpez.com>

# Linux

## Les derniers tutoriels

### Administration de NFS sous linux

Ce cours vous apprend comment installer et configurer un partage de fichiers avec NFS

Lire le cours de **coredump** ([lien 35](#))

### NTFS sous Linux

Ce tutoriel explique de manière très détaillée l'utilisation de partitions de type NTFS sous linux.

Lire le cours de **Richard Russon** ([lien 36](#))

## Critique de livre

### “/proc et /sys” de Olivier Daudel, Editions O'Reilly

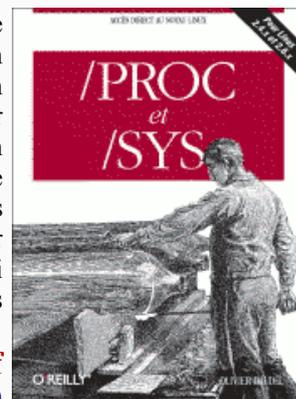
Ce livre s'adresse aux administrateurs de système Linux et aux développeurs qui veulent interagir directement avec le noyau Linux. Grâce à une multitude d'exemples, ce livre est un excellent support pour l'autoformation et ensuite grâce à l'index et aux nombreux tableaux qui le jalonnent, il deviendra une référence indispensable. Les histoires de Linux et d'Unix sont souvent évoquées pour permettre de comprendre comment on est

arrivé au noyau d'aujourd'hui, pourquoi certaines informations se trouvent à la fois disponibles dans /proc et dans /sys... Par ailleurs, l'auteur donne une multitude de liens qui permettent soit d'approfondir un sujet en particulier soit abordant un sujet évoqué mais sortant du cadre de l'ouvrage.

Comme souvent chez O'Reilly ce livre est vraiment d'une grande qualité, il aborde un des domaines qui font la richesse des

système Unix. Ce n'est pas le genre de livre que l'on lit d'un bloc et qui reste sur l'étagère, on y replonge régulièrement, pour éclaircir un point ou reprendre un passage survolé en première lecture. Un seul regret, je n'ai pas trouvé le moyen de télécharger les petits codes utilisés, ainsi qu'une liste des références sous forme électronique.

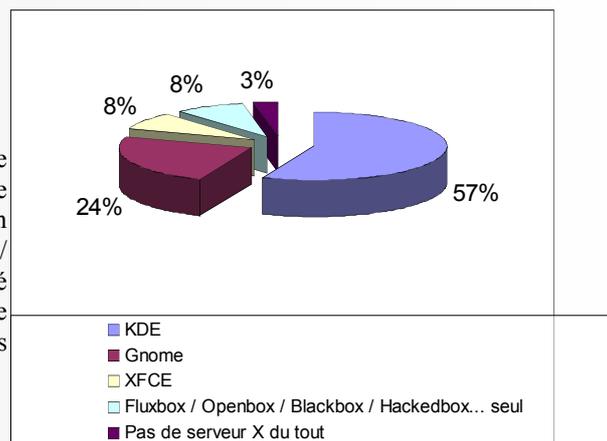
Critique du livre par **sinarf** ([lien 37](#))



## Sondage : votre environnement graphique

Le tuning : C'est important. Tout le monde l'admet. Le plaisir de configurer, de "thèmer" comme on dit son bureau, soit par un thème téléchargé, soit faire le sien! Quelque soit le WM, la liberté est très grande! Quel plaisir de montrer un super thème à ses amis lors d'un screenshot party, partant du vieux troll Kde/Gnome.. Du Bonheur! Un autre élément, qui ne m'a pas surpris, c'est la légèreté du WM. Les utilisateurs de Kde/Gnome, pourtant, ne semblent pas se plaindre d'avoir un WM assez lourd. Peut être KDE/Gnome propose des fonctionnalités lourdes mais intéressantes.

Retrouvez la rubrique Linux sur <http://linux.developpez.com>



# Liens

Vous trouverez ici la liste des URL des articles cités dans ce magazine. Pour des raisons pratiques de mise en page des URL longues, ces dernières sont fournies en fin de ce magazine. Ces URL sont actives : cliquez dessus pour accéder au document référencé.

1. <http://beuss.developpez.com/tutoriels/java/jakarta/commons/logging/>
2. <http://beuss.developpez.com/tutoriels/java/jakarta/commons/collections/>
3. <http://adiguba.developpez.com/tutoriels/java/tiger/annotations/>
4. [http://blog.developpez.com/index.php?blog=40&title=quoi\\_de\\_neuf\\_dans\\_java\\_se\\_6\\_mustang](http://blog.developpez.com/index.php?blog=40&title=quoi_de_neuf_dans_java_se_6_mustang)
5. [http://java.developpez.com/faq/java/?page=execution#signature\\_jar](http://java.developpez.com/faq/java/?page=execution#signature_jar)
6. <http://lagace.developpez.com/extreme-programming/>
7. <http://medina.developpez.com/cours/extreme-programming/>
8. <http://wpetrus.developpez.com/uml/mdaenaction/>
9. <http://goetter.developpez.com/bloc-inline/>
10. <http://sjrd.developpez.com/delphi/delphitutoriels/composants/nonvisuels/>
11. <http://laurent-dardenne.developpez.com/articles/Delphi/2005/DotNet/Enum-et-Enumbuilder/>
12. <http://flash-koti.developpez.com/articles/delphi/103-gestionexception/>
13. <http://jmdoudoux.developpez.com/delphi/menus/>
14. <http://jmdoudoux.developpez.com/delphi/asp/com/>
15. <http://jmdoudoux.developpez.com/delphi/fiches/>
16. <http://delphi.developpez.com/livres/#L2212113099>
17. <http://www.developpez.net/forums/viewforum.php?f=83>
18. <http://frenot.developpez.com/cours/extreme-programming/>
19. <http://java.developpez.com/sdjournal/tutoriel/temps-reel/>
20. <http://guigui.developpez.com/Tutoriel/Delphi/PythonD7BDD/>
21. <http://perl.developpez.com/tutoriel/cours1/>
22. <http://webman.developpez.com/articles/sqlserver/smo/>
23. <http://frechy.developpez.com/persmsn/>
24. <http://webman.developpez.com/articles/dotnet/gzip/>
25. <http://ditch.developpez.com/aspnet/tome5/>
26. <http://webman.developpez.com/articles/aspnet/masterpage/>
27. <http://dotnet.developpez.com/faq/delphi/>
28. <http://dotnet.developpez.com/FAQ.NET/>
29. <http://dotnet.developpez.com/faq/asp/>
30. <http://faqcsharp.developpez.com/>
31. <http://faqvbnet.developpez.com/>
32. <http://beaussier.developpez.com/articles/php/mysql/blob/>
33. <http://php.developpez.com/sdjournal/xul-pour-php/>
34. <http://php.developpez.com/sdjournal/framework/>
35. <http://coredump.developpez.com/nfs/>
36. <http://linux.developpez.com/cours/ntfslinux/>
37. <http://linux.developpez.com/livres/#L15>