

Developpez

Magazine

Edition de Mai Juin 2006.

Numéro 5.

Magazine en ligne gratuit.

Diffusion de copies conformes à l'original autorisée.

Directeur de la publication : Bobby Joe Lambert

Réalisation : Cédric Chatelain

Rédaction : la rédaction de Developpez

Contact : magazine@redaction-developpez.com

Index

Java	Page 2
Dot Net	Page 7
Systèmes	Page 10
Linux	Page 12
Delphi	Page 17
Web et dev. web	Page 20
XML et XMLRAD	Page 24
Oracle	Page 26
Access	Page 27
4D	Page 29
C et C++	Page 30
VB et VBA	Page 33
UML	Page 36
Liens	Page 38

Editorial

Developpez.com vous propose, depuis mi avril, des nouveaux forums. Avec encore plus de fonctionnalités, plus de facilités pour poser vos questions, grâce à la gestion de pièces jointes qui vous permet de partager des éléments avec les membres du club.

Et toujours le magazine que nous essayons de faire de plus en plus riche.

Tutoriel JAVA

Spring MVC par l'exemple



Cet article initie une série de 2 - 3 articles sur Spring MVC. Le modèle MVC (Modèle - Vue - Contrôleur) est maintenant bien ancré dans le monde du développement web en Java. Le framework Struts y est pour beaucoup. Il a amené une forme de standardisation du développement web MVC en Java.

par **Serge Tahé**

[Page 2](#)

Tutoriel Dot Net

Utiliser les références faibles sous .NET



*Dans un environnement managé comme Java ou .NET, tant qu'une référence existe sur un objet ce dernier ne peut pas, par définition d'un environnement managé, être libéré. Dilemme... C'est là qu'interviennent les références faibles, ou **Weak References**. Nous allons ici voir comment ces références particulières sont mises en oeuvre sous .NET à la fois sous C# et Delphi.NET.*

par **Olivier Dahan**

[Page 7](#)

Les derniers tutoriels et articles

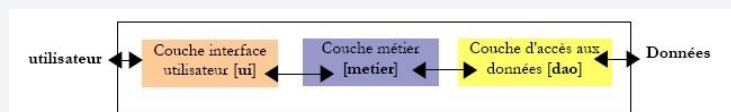
Spring MVC par l'exemple - partie 1

1. Généralités

Dans le développement web, la méthodologie de développement MVC (Modèle-Vue-Contrôleur) est désormais bien ancrée.

Rappelons-en le principe.

Une application web a souvent une architecture 3 tiers :



- la couche [**dao**] s'occupe de l'accès aux données, le plus souvent des données persistantes au sein d'un SGBD. Mais cela peut être aussi des données qui proviennent de capteurs, du réseau, ...

- la couche [**metier**] implémente les algorithmes " métier " de l'application. Cette couche est indépendante de toute forme d'interface avec l'utilisateur. Ainsi elle doit être utilisable aussi bien avec une interface console, une interface web, une interface de client riche. Elle doit ainsi pouvoir être testée en-dehors de l'interface web et notamment avec une interface console. C'est généralement la couche la plus stable de l'architecture. Elle ne change pas si on change l'interface utilisateur ou la façon d'accéder aux données nécessaires au fonctionnement de l'application.

- la couche [**interface utilisateur**] qui est l'interface (graphique souvent) qui permet à l'utilisateur de piloter l'application et d'en recevoir des informations.

La communication va de la gauche vers la droite :

- l'utilisateur fait une demande à la couche [**interface utilisateur**]
- cette demande est mise en forme par la couche [**interface utilisateur**] et transmise à la couche [**métier**]
- si pour traiter cette demande, la couche [**métier**] a besoin des données, elle les demande à la couche [**dao**]
- chaque couche interrogée rend sa réponse à la couche de gauche jusqu'à la réponse finale à l'utilisateur.

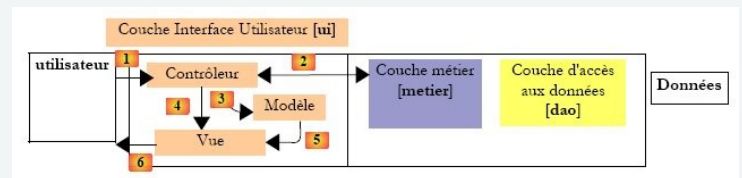
Les couches [métier] et [dao] sont normalement utilisées via des interfaces Java. Ainsi la couche [métier] ne connaît de la couche [dao] que son ou ses interfaces et ne connaît pas les classes les

implémentant. C'est ce qui assure l'indépendance des couches entre elles : changer l'implémentation de la couche [dao] n'a aucune incidence sur la couche [métier] tant qu'on ne touche pas à la définition de l'interface de la couche [dao]. Il en est de même entre les couches [interface utilisateur] et [métier].

L'architecture MVC prend place dans la couche [**interface utilisateur**] lorsque celle-ci est une interface web. Des articles (par exemple : <http://tahe.developpez.com/java/m2vc>) ont montré qu'on pouvait également appliquer le paradigme MVC à une couche

[**interface utilisateur**] Swing.

Au sein de l'architecture 3tier, l'architecture MVC peut être représentée comme suit :



Le traitement d'une demande d'un client se déroule selon les étapes suivantes :

1. le client fait une demande au contrôleur. Celui-ci voit passer toutes les demandes des clients. C'est la porte d'entrée de l'application. C'est le C de MVC.
2. le contrôleur C traite cette demande. Pour ce faire, il peut avoir besoin de l'aide de la couche métier. Une fois la demande du client traitée, celle-ci peut appeler diverses réponses. Un exemple classique est :
 - une page d'erreurs si la demande n'a pu être traitée correctement
 - une page de confirmation sinon
3. le contrôleur choisit la réponse (= vue) à envoyer au client. Choisir la réponse à envoyer au client nécessite plusieurs étapes :
 - choisir l'objet qui va générer la réponse. C'est ce qu'on appelle la vue V, le V de MVC. Ce choix dépend en général du résultat de l'exécution de l'action demandée par l'utilisateur.
 - lui fournir les données dont il a besoin pour générer cette réponse. En effet, celle-ci contient le plus souvent des informations calculées par le contrôleur. Ces informations forment ce qu'on appelle le modèle M de la vue, le M de MVC.L'étape 3 consiste donc en le choix d'une vue V et en la construction du modèle M nécessaire à celle-ci.

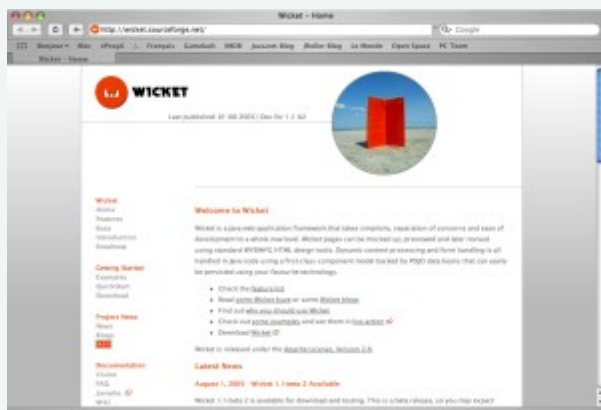
Retrouvez la suite de l'article de Serge Tahé en ligne [[Lien1](#)]
Et les autres parties de cet article [[Lien2](#)]

Redécouvrez le web avec Wicket

Java constitue une formidable plateforme de développement que cela soit pour les clients riches ou les applications web. S'il n'est pas nécessaire d'utiliser Java EE pour réaliser un site web, il est nécessaire de faire appel aux services d'un framework.

1. Introduction

Malgré son succès pour le déploiement d'applications web côté serveur, Java souffre d'une réputation de technologie compliquée et difficile à mettre en oeuvre. Cette constatation est parfaitement vraie dans le cas de Java Enterprise Edition, conçue pour soutenir de très grosses applications réparties sur plusieurs machines. La plateforme Java SE suffit cependant pour construire des applications web grâce aux servlets, aux Java Server Pages (JSP) et à un simple serveur comme Tomcat, également appelé conteneur de servlet. Les JSP et servlets ne constituent que la couche de plus bas niveau des applications web écrites en Java. Pour simplifier et accélérer les développements, de nombreux frameworks sont disponibles, proposant tous des fonctionnalités et des concepts très différents.



Le site de Wicket propose des liens vers les blogs des développeurs

Certains de ces frameworks, comme Spring, Turbine, Tapestry, Cocoon ou encore le fameux Struts, sont largement répandus et utilisés par de nombreuses entreprises. Bien que Wicket ne soit à première vue qu'un framework de plus dans cet univers déjà surchargé, les motivations de ses auteurs sont très claires. Contrairement aux offres existantes, Wicket est simple à apprendre, ne nécessite pas de code HTML spécifique ni de fichier de configuration XML compliqué et repose sur une API semblable à Swing. A l'instar des Java Server Faces (JSF) ou d'ASP.NET, Wicket maintient en outre automatiquement l'état des composants sur le serveur. Un champ texte d'une page HTML est par exemple lié à un composant et à un modèle Java du côté serveur. Cette solution permet de vous affranchir d'une grande partie de la gestion des sessions utilisateurs. Enfin, comme nous allons le voir, Wicket rend très facile la séparation entre la logique et la présentation.



Certains composants Wicket sont complexes, comme cet exemple d'arbre.

2. Mise en route

Le framework Wicket existe aujourd'hui en versions 1.0 et 1.1-beta2. Nous allons utiliser cette dernière pour nos premiers pas. Vous trouverez cet exemple dans l'application login-wicket sur le CD-Rom. La distribution binaire contient une bibliothèque appelée wicket-1.1-b2.jar que vous devrez copier dans le dossier WEB-INF/lib de votre application web. Vous devrez également copier les dépendances qui se trouvent dans le répertoire lib/ de Wicket. Il ne vous reste plus qu'à créer le descripteur web.xml de l'application comme celui présenté dans le listing 1.

Listing 1

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app
    PUBLIC "-//Sun Microsystems, Inc.//DTD Web
    Application 2.3//EN"
    "http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>
  <display-name>Login: Wicket</display-name>
  <servlet>
    <servlet-name>HelloWorldApplication</servlet-name>
    <servlet-
class>wicket.protocol.http.WicketServlet</servlet-class>
    <init-param>
      <param-name>applicationClassName</param-name>
      <param-
value>com.loginmag.HelloWorldApplication</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>HelloWorldApplication</servlet-name>
    <url-pattern>/helloworld/*</url-pattern>
  </servlet-mapping>

  <servlet>
    <servlet-name>ChatApplication</servlet-name>
    <servlet-
class>wicket.protocol.http.WicketServlet</servlet-class>
    <init-param>
      <param-name>applicationClassName</param-name>
      <param-value>com.loginmag.ChatApplication</param-
value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>ChatApplication</servlet-name>
    <url-pattern>/chat/*</url-pattern>
  </servlet-mapping>
</web-app>
```

Ce descripteur peut prêter à confusion car il est possible de créer plusieurs applications Wicket dans une même application web. Pour créer une nouvelle application Wicket, il vous suffit d'utiliser le servlet WicketServlet et d'utiliser le paramètre applicationClassName pour indiquer la classe représentant l'application à proprement parler. Une application est une classe Java héritant wicket.protocol.http.WebApplication et servant de point d'entrée et de configuration. La méthode **getPages()** renvoie la propriété la plus importante de votre application, une instance d'ApplicationPages qui vous permet de définir la page d'accueil et les pages d'erreur. Le listing 2 présente l'implémentation minimale d'une application Wicket.

listing 2

```
package com.loginmag;

import wicket.protocol.http.WebApplication;

public class HelloWorldApplication extends
WebApplication {
    public HelloWorldApplication() {
        getPages().setHomePage(HelloWorldPage.class);
    }
}
```



Suivant l'API Swing au plus près, Wicket propose une méthode `paint()` pour générer des images.

La page d'accueil est passée en paramètre de la méthode `setHomePage()` sous la forme d'une classe Java (une instance de Class), dans notre cas `HelloWorldPage.class`. Une fois la page d'accueil spécifiée, toute requête sur l'URL `http://localhost:8080/login-wicket/helloworld` renverra le code HTML généré par la page `HelloWorldPage`.

Une page Wicket est constituée de deux éléments : une classe Java héritant `wicket.markup.html.WebPage` et un fichier HTML portant le même nom que la classe, `HelloWorldPage.html` dans notre cas. Le listing 3 présente le contenu de la page HTML.

Listing 3

```
<html>
  <head>
    <title>Login: Wicket</title>
  </head>
  <body>
    <span wicket:id="message">Emplacement du
message</span>
  </body>
</html>
```

Le code n'utilise que des balises HTML ou XHTML classiques. La seule différence réside dans l'utilisation du namespace `wicket` pour identifier les composants pour le code Java. Dans cet exemple nous avons créé un élément `` appelé `message` et contenant un texte par défaut.



Tomcat est un conteneur de servlet gratuit et open source très simple à utiliser avec Wicket.

Cette approche, très simple, permet aux designers de concevoir l'apparence visuelle du site sans se soucier d'intégrer des instructions de programmation. Ils pourront ainsi éditer les pages dans un outil comme DreamWeaver ou Nvu, à condition de conserver les `wicket:id` des composants. Lors de la conception visuelle des pages, les designers utiliseront les valeurs par défaut, comme celle de notre exemple. De la même manière, les développeurs n'auront pas à s'inquiéter d'interférer avec l'apparence du site et pourront se concentrer sur l'implémentation de la logique. Pour vous en convaincre, étudiez le code Java de `HelloWorldPage` présenté dans le listing 4.

Listing 4

```
package com.loginmag;

import wicket.markup.html.WebPage;
import wicket.markup.html.basic.Label;

public class HelloWorldPage extends WebPage {
    public HelloWorldPage() {
        add(new Label("message", "Hello World !"));
    }
}
```

L'unique ligne de cette page ajoute un composant de type `Label` pour lequel le premier paramètre définit le nom et le second paramètre la valeur. Lorsqu'un visiteur arrivera sur votre page, Wicket exécutera votre code pour remplacer le contenu de la balise `` par la valeur du `Label`.



Les spécifications de Java EE 5 font plus de 300 pages... Wicket est bien plus simple !

3. Utilisation des formulaires

Notre second exemple est un petit chat que vous trouverez dans l'application Wicket ChatApplication. Il ne contient qu'une seule page, ChatPage, qui permet aux visiteurs d'envoyer des messages. Le listing 5 contient une version simplifiée de la page HTML représentant le chat.

Listing 5

```
<form wicket:id="chatForm">
  Surnom : <input type="text"
wicket:id="nick"></input><br />
  Message : <input type="text"
wicket:id="message"></input><br />
  <input type="submit" value="Envoyer" />
</form>
<span wicket:id="messages">
  <span wicket:id="nick">Invité</span> : <span
wicket:id="message">Message.</span><br /><br />
</span>
```

Dans cette page se trouvent deux composants racine, un formulaire chatForm et une liste de message appelée messages. Ils contiennent chacun un champ nick et un champ message pour permettre au visiteur de saisir ou de voir les messages. En aparté, cet exemple montre que Wicket gère parfaitement des arbres de composants. Nous avons appris au début de cet article que les composants HTML sont liés à un composant Java et à un modèle. Dans cet exemple, le formulaire permet de remplir les informations d'un modèle qui seront lues pour l'affichage des messages. A l'opposé de Java EE et de ses EJB, Wicket se contente de POJO (Plain Old Java Object) pour gérer les modèles. Le modèle du chat correspond à la classe Message qui se trouve sur le CD-Rom.

La réalisation d'un chat en Wicket ne nécessite que quelques lignes de code. Elle implémente simplement l'interface Serializable pour permettre à Wicket de la sauvegarder en session, et expose ses attributs avec des accesseurs et mutateurs. Conformément à la page HTML, un message contient le surnom de l'utilisateur, attribut nick, et son message, attribut message. La réalisation de la page Java responsable de la liaison entre la page HTML et le modèle demande d'utiliser de nouveaux composants Wicket comme le montre le listing 6 qui contient la définition de la page ChatPage.

Listing 6

```
public class ChatPage extends WebPage {
  private static List messagesList = new ArrayList();
  private ListView messagesListView;

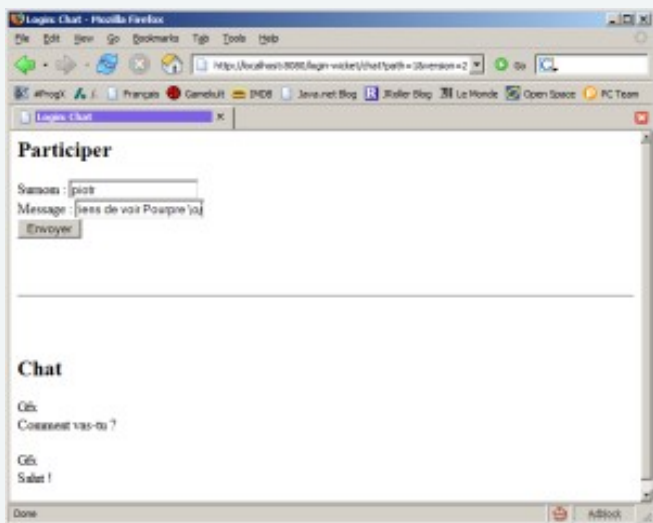
  public ChatPage() {
    add(new ChatForm("chatForm"));
    add(messagesListView = new ChatListView("messages",
messagesList));
  }
  // ...
}
```

Le premier composant personnalisé est ChatForm, dérivant de Form, capable de gérer les champs du formulaire. Le second composant est un dérivé de ListView, ChatListView, que nous utilisons pour afficher la liste des messages saisis par les utilisateurs. Le modèle de cette liste est une ArrayList statique afin qu'elle soit visible de tous les visiteurs du site. Lors du rendu de la page, la méthode *populateListItem()* du composant ChatListView sera invoquée pour chaque élément de la liste messagesList. Le listing 7 présente le code utilisé pour modifier les valeurs des composants HTML correspondant à chaque élément.

Listing 7

```
private class ChatListView extends ListView {
  // ...
  public void populateItem(ListItem listItem) {
    Message message = (Message)
listItem.getModelObject();
    listItem.add(new Label("nick", message.getNick()));
    listItem.add(new Label("message",
message.getMessage()));
  }
}
```

Retrouvez la suite de l'article de Guy Romain en ligne [[Lien44](#)]



Compte Rendu Javapolis 2005

Javapolis 2005 s'est tenu du 12 au 16 décembre 2005. L'équipe Java de developpez.com s'y est rendue. En voici le compte rendu.



1. Introduction

L'édition Javapolis 2005 n'a pas déçue. Son fondateur principal, Stephan Janssens, également fondateur de BEJUG, le Java User Group pour la Belgique, a voulu faire de Javapolis LE salon Java pour le continent européen. Et on peut dire qu'il a réussi puisque des visiteurs de tous les pays d'Europe étaient représentés. Seul le Portugal était absent.

Mais, contrairement à JavaOne, Javapolis se veut abordable. En effet, le prix d'entrée pour toute la semaine ne s'élevait qu'à 300€. Et je dois que pour ce prix là, on ne se fait pas voler. La seule ombre au tableau fut la connection Wi-Fi, qui rendait quasiment impossible l'accès à internet pour la plupart d'entre nous, sauf alors à des prix exorbitants. Mais Stephan Janssens a promis de solutionner cela pour les

prochaines éditions.

1.1. En a-t-on pour son argent ?

Combien devriez-vous payer pour avoir un cours XML d'une demi journée ?

Et pour un cours d'une journée sur Spring ?

Et une introductions aux EJB ?

Vous avez vite fait de convaincre votre employeur qu'il a tout intérêt à vous inscrire à Javapolis, pour que vous puissiez aller suivre bien plus que cela, pour beaucoup moins cher.

De plus, vous recevez un magnifique t-shirt rouge, au logo de Javapolis,



un magnifique sac à dos, aux couleurs de Javapolis, contenant un bic, un carnet de note, le programme de la semaine et des dépliants de certains partenaires.

Boissons à volonté durant tout le salon. Qu'elles soient froides ou chaudes. Des couques et croissants le matin, un sandwich à midi. (On a même eu droit à un repas chaud un jour (des nouilles)). Sans oublier le film de la semaine (cette année, ce fut King-Kong).

2. Les vedettes du salon

2.1. Au niveau des conférences

Si je devais, personnellement, retenir qu'une seule chose de ce salon, c'est la suivante :

Java EE 5 fut la vedette incontestée de ce salon. Que ce soit :

- via EJB3 (qui va simplifier grandement le développement des EJB),
- via JPA (enfin une API standard pour la persistance),

- Glassfish (l'implémentation de référence Open Source de Java EE 5),
- NetBeans EE 5.0 (EDI prêt entre autre pour le développement Java EE 5),
- Oracle TopLink (qui implémente l'API JPA et qui est intégré dans Glassfish),

Java EE 5 était mentionné tous les jours de cette semaine.

Retrouvez la suite de l'article de Vincent Brabant en ligne [[Lien3](#)]

Vu sur les Blogs

les Forums Java de Developpez en RSS

Avec la nouvelle version du forum de developpez.com (j'imagine que vous aviez remarqué 😊), la grosse nouveauté, c'est les flux Rss par forum. Du coup vous pouvez afficher dans votre application préférée de syndication les différents forums java.

Pour cela il vous suffit d'utiliser une url de la forme suivante <http://www.developpez.net/forums/external.php?type=rss2&forumids=X> en remplaçant X par le numéro du forum que vous désirez.

- [...forumids=22](#) pour Général Java
- [...forumids=114](#) pour Interfaces Graphiques Java

- [...forumids=61](#) pour J2EE
- [...forumids=94](#) pour Eclipse
- [...forumids=31](#) pour JBuilder
- [...forumids=115](#) pour NetBeans & Sun Java Studio Creator



Vous pouvez même combiner ces différents forums en ajoutant les numéros de ces forums à la fin de l'url séparés d'une virgule.

Par exemple [...forumids=22,114](#) vous agrège les contenus de Général java et Interfaces Graphiques.

Et bien sur, ça marche pour tous les forums de developpez. Bon netvibes/googleIg/RssOwl,thunderbird ou autre !

Retrouvez le blog de Lunatix en ligne [[Lien4](#)]

Vu sur Developpez.tv

Les vidéos de Code Camp 2006

[Sun France](#)[[Lien5](#)] a organisé à Paris, le 5 et 6 avril 2006, deux journées sur Java et Solaris. La première journée était faite de présentations théoriques, la seconde de travaux pratiques. Nous vous proposons ici la rediffusion des présentations de la première journée.

Les Vidéos sont au format DivX, en 320x240.

N'hésitez pas à nous laisser vos [commentaires](#)[[Lien6](#)]

Bon visionnage !!! Retrouvez les vidéos de Code Camp 2006 en ligne [[Lien7](#)]



Utiliser les références faibles sous .NET

Dans un environnement managé comme .NET la gestion des références à des instances de classe semble naturelle et sans poser de souci. Un objet est soit référencé, donc « vivant », soit n'est plus référencé, donc « mort » et éligible à sa destruction par le Garbage Collector (plus loin noté GC). Si cela correspond le plus souvent au besoin, il est des cas où l'on voudrait conserver une référence sur un objet sans pour autant interdire son éventuelle libération. C'est le cas d'un cache d'objets par exemple : il référence des objets et peut les servir si besoin est, mais si la mémoire manque et que certains ont été détruits entre temps, cela n'est pas grave, ils seront recréés. Or, dans un environnement managé comme Java ou .NET, tant qu'une référence existe sur un objet ce dernier ne peut pas, par définition d'un environnement managé, être libéré. Dilemme... C'est là qu'interviennent les *références faibles*, ou **Weak References**. Nous allons ici voir comment ces références particulières sont mises en œuvre sous .NET à la fois sous C# et Delphi.NET.

1. Définition

Une référence faible est une référence à un objet qui bien qu'elle en autorise l'accès n'interdit pas la possible suppression de ce dernier par le Garbage Collector.

En clair, cela signifie qu'une référence faible, d'où sa « faiblesse », ne crée pas un lien fort avec l'instance référencée. C'est une référence qui, du point de vue du système de libération des instances n'existe pas, elle n'est pas prise en compte dans le graphe des chemins des objets « accessibles » par le GC.

C'est en fait comme cela que tout fonctionne en POO classique dans des environnements non managés comme Win32, rien n'interdit une variable de pointer un objet qui a déjà été libéré, mais tout accès par ce biais se soldera par une violation d'accès.

Sous environnement non managé il n'existe pas de solution simple pour éviter de telles situations, d'où l'engouement croissant pour les environnements managés comme .NET ou Java sans qu'aucun retour en arrière ne semble désormais possible, n'en déplaise à ceux qui restent attachés à Win32.

En effet, sous .NET une telle situation ne peut tout simplement pas arriver puisqu'on ne libère pas la mémoire explicitement, c'est le CLR qui s'en charge lorsqu'il n'y a plus de référence à l'objet.

Il ne faut d'ailleurs pas confondre mémoire et ressources externes. .NET protège la mémoire, pas les ressources externes. Pour cela les classes doivent implémenter `IDisposable`. Et si un objet a été « disposé » il n'est pas « libéré » pour autant. On peut donc continuer à l'utiliser mais cela créera le plus souvent une erreur d'exécution puisque les ressources externes auront été libérées entre temps... Prenez une instance de `System.Drawing.Font`, appelez sa méthode `Dispose()`. Vous pourrez toujours accéder à l'objet en tant qu'entité, mais si vous tenter d'appeler sa méthode `ToHFont()` qui retourne le handle de l'objet fonte sous-jacent, une exception sera levée... Il existe une nuance importante entre mémoire et ressources externes, entre libération d'un objet et libération de ses ressources externes. C'est là l'une des difficultés du modèle objet de .NET qui pose souvent des problèmes aux débutants, et parfois même à des développeurs plus confirmés.

2. Les mécanismes en jeu

Le Garbage Collector du CLR libère la mémoire de tout objet qui ne peut plus être atteint. Un objet ne peut plus être atteint quand toutes les références qui le pointent deviennent non valides, par exemple en les forçant à null (nil sous Delphi). Lorsqu'il détruit les objets qui se trouvent dans cette situation le GC appelle leur méthode `Finalize`, à condition qu'une telle méthode soit définie et que le GC en ait été informé (le mécanisme réel est plus complexe

et sort du cadre de cet article).

Lorsqu'un objet peut être directement ou indirectement atteint il ne peut pas être supprimé par le GC. Une référence vers un objet qui peut être atteint est appelée une référence forte.

Une référence faible permet elle aussi de pointer un objet qui peut être atteint qu'on appelle la cible (*target* en anglais). Mais cette référence n'interfère pas avec le GC qui, si aucune référence forte n'existe sur l'objet, peut détruire ce dernier en ignorant les éventuelles références faibles (elles ne sont pas totalement ignorées puisque, nous allons le voir, la référence faible sera avertie de la destruction de l'objet).

Les références faibles se définissent par des instances de la classe **WeakReference**. Elle expose une propriété `Target` qui permet justement de réacquérir une référence forte sur la cible. A condition qu'il existe encore... C'est pour cela que cette classe offre aussi un moyen de le savoir par le biais de sa propriété `IsAlive` (« est il encore vivant ? »).

Pour un système de cache, comme évoqué en introduction, cela est très intéressant puisqu'on peut libérer des objets (plus aucune référence valide ne le pointe) et malgré tout le récupérer dans de nombreux cas si le besoin s'en fait sentir. Cela est possible car entre le moment où un objet devient éligible pour sa destruction par le GC et le moment où il est réellement collecté et finalisé il peut se passer un temps non négligeable !

Le GC utilise trois « générations », trois conteneurs logiques. Les objets sont créés dans la génération 0, lorsqu'elle est pleine le GC supprime tous les objets inutiles et déplace ceux encore en utilisation dans la génération 1. Si celle-ci vient à être saturée le même processus se déclenche (nettoyage de la génération 1 et déplacement des objets encore valides dans la génération 2 qui représente toute la RAM disponible).

De fait, un objet qui a été utilisé un certain temps se voit pousser en génération 2, un endroit qui est rarement visité par le GC. Parfois même, s'il y a beaucoup de mémoire installée sur le PC et si l'application n'est pas très gourmande, les objets de la génération 2, voire de la génération 1, ne seront jamais détruits jusqu'à la fermeture de l'application... A ce moment précis le CLR videra d'un seul coup tout l'espace réservé sans même finaliser les objets ni appeler leur destructeur. C'est pourquoi sous .NET on ne programme généralement pas de destructeurs dans les classes : le mécanisme d'appel à cette méthode n'est pas déterministe.

Donc, durant toute la vie de l'application de nombreuses instances restent malgré tout en vie « quelque part » dans la RAM. Si une référence faible pointe l'un de ces objets il pourra donc être « récupéré » en réacquérant une référence forte dessus par le biais de

la propriété Target de l'objet WeakReference. Si l'objet en question réclame beaucoup de traitement pour sa création, il y a un énorme avantage à le récupérer s'il doit resservir au lieu d'avoir à le recréer, le tout sans pour autant engorger la mémoire puisque, si nécessaire, le GC l'aura totalement libéré, ce qu'on saura en interrogeant la propriété IsAlive de l'objet WeakReference qui aura alors la valeur false.

Si vous vous souvenez de ce que nous disions plus haut sur la nuance entre libération d'une instance et libération de ses ressources externes, vous comprenez que l'application ne doit utiliser des références faibles que sur des objets qui n'implémentent pas IDisposable. En effet, pour reprendre l'exemple d'une instance de la classe Font, si avant de mettre la référence à null votre application a appelé sa méthode Dispose(), « récupérer » plus tard l'instance grâce à une référence faible sera très dangereux : les ressources externes sont déjà libérées et toute utilisation de l'instance se soldera par une exception. Il est donc important de se limiter à des classes non disposables.

3. L'intérêt

Les références faibles ne servent pas qu'à mettre en œuvre des systèmes de cache, elles servent aussi lorsqu'on doit pointer des objets qui peuvent et doivent éventuellement être détruits. Rappelons-nous : si nous utilisons une simple référence sur un tel objet, il ne sera jamais détruit puisque justement nous le référençons... Les références faibles permettent d'échapper à ce mécanisme par défaut qui, parfois, devient une gêne plus qu'un

avantage.

Un exemple d'une telle situation : Supposons une liste de personnes. Cette liste pointe donc des instances de la classe **Personne**. Imaginons maintenant que l'application autorise la création de « groupes de travail », c'est-à-dire des listes de personnes. Si les listes définissant les groupes de travail pointent directement les instances de Personne et si une personne est supprimée de cette liste, les groupes de travail continueront de « voir » cette personne puisque l'instance étant référencée (référence forte) elle ne sera pas détruite par sa simple suppression de la liste des personnes...

En fait, on souhaitera dans un tel cas que toute personne supprimée de la liste principale n'apparaisse plus dans les groupes de travail dans lesquels elle a pu être référencée.

Cela peut se régler par une gestion d'événement : toute suppression de la liste des personnes entraînera le balayage de tous les groupes de travail pour supprimer la personne. Cette solution n'est pas toujours utilisable. Les références faibles deviennent alors une alternative intéressante, notamment parce qu'il n'y a pas besoin d'avoir prévu un lien entre la liste principale et les listes secondaires qui peuvent être ajoutées après coup dans la conception de l'application et parce que la suppression d'une personne n'impose pas une attente en raison du balayage de toutes les listes secondaires.

Retrouvez la suite tutoriel d'Olivier Dahan en ligne [[Lien8](#)]

ASP.NET 2.0: Implémenter son propre Membership provider en 30 minutes

1. Historique

Il y a quelques années, nous n'avions que ASP pour réaliser nos applications. Il fallait dès lors écrire toutes les méthodes pour gérer les utilisateurs au travers de l'application.

Ensuite, avec l'arrivée d'asp.net 1.x, cette gestion a été simplifiée grâce à l'arrivée du mode d'authentification "forms". Malheureusement il restait pas mal de code à écrire...

Asp.net 2.0 amène toute une série de nouveautés concernant la gestion des utilisateurs. On y retrouve :

- des providers (la couche d'accès aux données en quelque sorte)
- des composants (la partie visuelle du membership)
- des API

On retrouve aussi les différentes couches nécessaires à l'implémentation d'une gestion d'utilisateurs, ce qui nous permet de nous concentrer sur le cœur de l'application que nous réalisons.

Nous allons dès lors présenter chacune des couches.

2. Les "API Membership"

A quoi servent les API Membership? Elles permettent de :

- Créer des utilisateurs
- Sauvegarder des informations dans des environnements différents
- Authentifier des utilisateurs
- Gérer des mots de passe

Quel est l'intérêt d'utiliser ces API? Par définition il s'agit de faire l'interface entre les applications et les sources de données.

De plus, asp.net 2.0 est "prêt" à utiliser ces différentes API. Effectivement, tous les composants "Login", comme LoginName, Login, CreateUser, ... utilisent ces API.

3. Membership service



Sur ce schéma, on peut y voir plusieurs couches (de bas en haut) :

- Sources de données
- Providers
- APIs
- Contrôles de login

Sources de données

Il s'agit des bases de données, fichiers, Active Directory, ...

Providers

Chaque provider doit permettre d'accéder à une source de données en particulier. Il fournit les différentes méthodes pour écrire / lire dans cette source. Nous allons détailler les providers un peu plus loin dans cet article.

APIs

Les APIs sont des classes qui possèdent différentes méthodes qui permettent d'effectuer certaines actions. Ces méthodes s'appuient sur le provider en action.

Contrôles de login

Alors qu'il était souvent nécessaire de réaliser son propre contrôle pour authentifier les utilisateurs ou encore manipuler les informations s'y rapportant, asp.net 2.0 a amené toute une série de contrôles le permettant.

Retrouvez la suite du tutoriel de Ditch en ligne [[Lien9](#)]

Protégez, optimisez et contrôlez votre code avec {smartassembly}

Découverte et présentation du logiciel {smartassembly}, un must-have pour tout développeur, qui permet d'optimiser, obfusquer et déboguer son code, le plus simplement et le plus efficacement possible.

1. Introduction

Chaque jour, de très nombreux programmes voient le jour à l'aide des nouveaux langages. Néanmoins, en sus de la politique sur les brevets logiciels, le respect des droits d'auteur est toujours aussi présent et il nécessite de pouvoir, si on le désire, protéger son code source. Ceci se pose notamment sous Dotnet, où le principe de langage intermédiaire permet de récupérer totalement le code source d'origine. Le seul moyen de se protéger de ce type de piratage est d'obscurcir ("obscurcir" : rendre illisible) son code. C'est en cela qu'entrent en jeu ces logiciels bien particuliers que sont les obfuscateurs.



Il existe deux logiciels très connus pour lire le code intermédiaire (Common Intermediate Language): ILDASM et [Reflector \[Lien10\]](#). L'utilisation de {smartassembly}, vous permettra de vous protéger du vol de code via ces logiciels. Attention, cette solution n'est pas parfaitement sûre, cependant elle permet de mieux protéger le code qu'il ne l'est par défaut.

2. Présentation de {smartassembly}

{smartassembly}, développée par la société Cachupa, vient de sortir dans sa version 1.0. Cette dernière est une application qui

permet à la fois de gérer la protection de votre code, l'optimisation de celui-ci et la gestion des exceptions lors de son utilisation.

Ses fonctionnalités sont les suivantes:

- fusion des assemblies
- - obfuscation du code
- - obfuscation des chaînes de caractères
- - suppression du code inutile
- - ajout d'une gestion des exceptions

Ce sont ces fonctionnalités que je vais tenter de vous présenter le plus clairement possible.

3. Utilisation

Des nombreux logiciels d'obfuscation que j'ai utilisé (Salamander, Xenocode, etc), {smartassembly} est au premier abord (et par la suite également), l'application la plus user-friendly, la plus claire et la plus simple à utiliser. L'interaction de l'utilisateur peut être limitée au minimum si ce dernier en a le désir. Nous allons découvrir les différentes fonctionnalités au fil des différentes étapes de l'utilisation de {smartassembly}.

Etrouvez la suite de l'article de Louis-Guillaume Morand en ligne [\[Lien11\]](#)

Vu sur les Blogs

Précisions sur les dates de sortie des Service Pack pour Visual Studio

Je vous annonçais [ici \[Lien12\]](#) que les Services Pack de Visual Studio (version 2003 et 2005) seraient bientôt disponibles 😊

Nous avons maintenant, grâce à [Jihad DANNAWI \[Lien14\]](#) (Microsoft France), des dates plus précises quant à cette disponibilité:

- **Visual Studio 2003 SP1**: Prévu pour Juin 06
- **Visual Studio 2005 SP1** : Prévu pour Septembre 06



Au programme, on retrouve des corrections de bugs, des fix, etc...

Bref, les gars de chez Microsoft ont plutôt bien bossé 😊

A+

Retrouvez le blog de Thomas Lebrun en ligne [\[Lien18\]](#)

Vu sur Developpez.tv

Visionnez les Microsoft Devdays 2006 comme si vous y étiez !

Chaque année Microsoft organise à travers la France les DevDays. Ces rencontres sont l'occasion pour les développeurs de découvrir les nouveautés de chez Microsoft.

Cette année, les DevDays coïncident avec la sortie des très attendus Visual Studio 2005, SQL Server 2005, et traitent bien sûr d'autres sujets comme Office 12, Atlas...

Les vidéos que nous vous proposons ont été filmées par [Maxence Hubiche \[Lien16\]](#) le 23 Mars 2006. Vidéos au format Windows Media Video.

Laissez-nous [vos commentaires ici \[Lien17\]](#) !

Bon visionnage !

Retrouvez les vidéos sur [developpez.tv \[Lien18\]](#)

Les derniers tutoriels et articles

Adaptabilité et tolérance aux fautes : intérêt des intergiciels réflexifs face à l'évolutivité des systèmes informatiques

Les systèmes informatiques d'aujourd'hui sont utilisés dans des environnements en constante évolution, et nécessitent par conséquent des implémentations particulièrement ' malléables ', notamment capables de reconfiguration ' à chaud '. L'utilisation de tels systèmes dans des contextes critiques pose cependant le problème de leur tolérance aux fautes. La réflexivité semble apporter une solution intéressante à ce problème, en permettant l'auto-représentation et l'auto-modification d'un système informatique. Dans cet article, nous nous intéressons à l'utilisation de la réflexivité pour la tolérance aux fautes de systèmes hautement adaptables, et notamment la répartition des mécanismes de réflexivité dans les différentes couches du support d'exécution (noyau, intergiciel, etc...).

1. Introduction

Depuis plusieurs années, il est classique de souligner l'utilisation de plus en plus massive de briques logicielles standardisées et réutilisables pour la construction de nouveaux systèmes informatiques. Il n'est que de citer le succès des composants sur étagères, COTS en anglais (« *Commercial Off The Shelf* »), ou celui du source libre pour témoigner de cette tendance. Il est cependant des marchés informatiques qui du fait de leurs spécificités n'ont été touchés que plus récemment par cette évolution de fond. Le marché des applications embarquées, à fort besoin de sûreté de fonctionnement, est de ceux-là. L'utilisation de

composants du marché pour de telles applications pose en effet problème, aussi bien pour le matériel que le logiciel. Les enjeux aussi bien économiques qu'humains exigent la mise en place de

mécanismes de tolérance aux fautes, indépendamment des briques réutilisées, pour durcir leur robustesse et exclure tout scénario catastrophe.

La réflexivité logicielle fait partie des approches qui ont été proposées pour résoudre ce défi technologique. Dans cet article, après avoir dans une première partie rappelé les notions de tolérance aux fautes et de réflexivité, nous montrons dans une seconde partie pourquoi la réflexivité doit être tendue pour englober dans un même cadre conceptuel l'ensemble des couches d'un système. Nous détaillons enfin dans une dernière partie la méthodologie que nous avons mise en place et les premiers résultats que nous avons obtenus. Nous concluons sur les perspectives de notre travail.

Retrouvez la suite de l'article de François Taiani en ligne [[Lien19](#)]

CPL : Courant porteur en ligne

Le CPL (Courant Porteur en Ligne) appelé aussi PLC (Power Line Communication) ou BPL (Broadband over Power Line) est une technologie qui commence à faire parler d'elle. Pour s'en convaincre il suffit de regarder le nombre d'articles qui sortent sur ce sujet depuis 2003 (Le monde informatique, 01Net, réseaux&telecoms, New York Times ...). Mais que se cache-t-il derrière cette « nouvelle » technologie ? Est-ce un chemin d'avenir à prendre ? En étudiant le CPL on s'aperçoit vite de certaines limites, des voix s'élèvent contre son utilisation. Pourquoi tout ce remue-ménage ? Ce document est axé essentiellement sur le CPL coté Hautes Fréquences à des fins de communication réseaux et non Basses Fréquences pour la domotique.

1. Le CPL c'est quoi ?

1.1. Définition

Le CPL est une technique qui permet d'utiliser les lignes électriques basse et moyenne tension (220 volts ou 380 Volts), pour y faire passer des ondes courtes à hautes fréquences sur la bande des 1,6 MHz à 30 MHz au moyen d'un couplage avec les signaux électriques (50Hz en France).

1.2 Historique

Cette technologie existe depuis les années 1980 comme méthode de transport des informations à bas débits pour des applications de domotique notamment pour piloter à distance des appareils électriques (radiateurs, lumière ...).

D'ailleurs, EDF (Electricité De France) l'utilise à cette époque pour effectuer ses maintenances à distance. Les plus répandues, qui fonctionnent toujours sont connues sous les appellations X10, Lonworks et CEBus.



Les modules émetteurs transmettent les signaux de commandes à une fréquence de 120 KHz sous une tension de 2.5V sur les mêmes câbles que le courant 230V.- 50 Hz. Pour éviter toutes confusions avec d'éventuels parasites, le signal est transmis 3 fois de suite à 3.33 milli-secondes d'intervalle.

Retrouvez la suite de l'article de ... en ligne [[Lien20](#)]



SSH, le shell sécurisé

La référence

540 pages, 1ère édition, janvier 2002 Editions O'REILLY, ISBN: 2-84177-147-4

Public visé : *expert*



Commander sur Amazon.fr

SSH (Secure SHell) est une application réseau basée sur TCP/IP qui permet de se connecter sur une machine distante et d'effectuer des transferts de fichiers entre machines en toute sécurité. SSH dispose en outre d'une fonctionnalité de tunneling unique, qui encapsule les données habituellement échangées en clair par les applications sur le réseau. Avec un chiffrement fort et transparent, une authentification par clé publique fiable et une architecture client/serveur hautement configurable, SSH assure sécurité et confidentialité. Quel que soit le type de réseau envisagé, SSH offre une connexion extrêmement sûre et robuste, tout en restant simple d'emploi. Le protocole SSH, qui existe sous deux versions et se décline en plusieurs implémentations libres ou commerciales, est compatible avec la plupart des systèmes d'exploitation. Écrit par des spécialistes du sujet, cet ouvrage traite en détail du shell sécurisé et démythifie les pages de manuels de SSH.

- SSH1, SSH2, OpenSSH et F-Secure SSH pour Unix, ainsi que les produits pour MS-Windows et Mac OS ;
- la configuration des serveurs et des clients SSH, tant au niveau système qu'au niveau de l'utilisateur, avec des conseils de configurations pour maximiser la sécurité ;
- la gestion avancée des clés à l'aide des agents, le transfert d'agent et les commandes forcées ;
- le transfert (tunneling) des applications TCP et X11, même en présence de pare-feux et de traduction des adresses NAT (masquerading) ;
- l'intégration de SSH avec Kerberos, PGP, PAM et d'autres logiciels liés à la sécurité ;
- les aspects non documentés des implémentations SSH les plus connues ;
- l'installation et la maintenance des systèmes SSH ;
- la résolution de différents problèmes, classiques et moins connus.

Que vous communiquiez sur un petit réseau local ou sur l'Internet (par exemple, si vous vous connectez fréquemment depuis l'Internet à un réseau privé), que vous soyez administrateur système ou utilisateur final, ce livre est fait pour vous !

Comme l'indique son titre, ce livre est **LA bible de SSH**. Les auteurs ont réussi l'exploit de s'adresser à un large public non spécialiste pour leur expliquer simplement l'usage de SSH pour des opérations courantes tout en leur inculquant les aspects de sécurité généraux qui soutendent l'utilisation de cet outil.

Critique du livre par la rédaction ([cyberzoïde](#)) :

Les experts et administrateurs systèmes ne sont pas en reste car cet ouvrage leur est principalement destiné : leur est décrit comment tirer parti au mieux de SSH. Outre des rappels nécessaires aux concepts de sécurité et de chiffrement employés avec SSH, toutes les fonctions de SSH sont passées en revues : tunneling, déport X, copies sécurisés scriptés, identités multiples...

Les auteurs ont également pensé aux développeurs qui voudraient intégrer la communauté et enrichir SSH.

Toute la doc et plus encore... Découvrez les fonctionnalités non documentées de SSH ! Mais aussi une FAQ, la page de manuel, un guide de référence rapide, les autres implémentations de SSH et les principaux clients des principales plateformes... Bref, tout pour une prise en main rapide.

Toujours avec le soucis d'exactitude et de clareté, les auteurs ont rassemblé dans ce livre toute l'information à l'usage des architectes, développeurs et ingénieurs systèmes. Il est difficile de tout assimiler d'une traite, mais ce n'est pas son ambition ; selon votre profil, reportez vous aux chapitres qui vous sont utiles : ils seront déjà bien suffisamment fournis !

C'est la référence que tout ingénieur système doit avoir à porté de main sur le bureau. Excellent!

Vu sur le dico

WiMAX

Worldwide Interoperability for Microwave Access

 *acr.*

Famille de normes de réseau [sans fil](#) créé initialement par les sociétés Intel Corporation et Alvarion en 2002. Elle a depuis fait l'objet d'une ratification par l'[IEEE](#) pour devenir le [IEEE 802.16](#).

WiMAX est désormais porté par le [WiMAX Forum](#) qui délivre le label [WiMAX Forum Certified](#) aux produits et services conformes à la norme.

Le WiMAX offre une connexion haut débit sans fil sur une longue distance pouvant être utilisée dans le cadre de la [boucle locale radio](#). La portée théorique est de 50 km pour un [débit](#) ([montant](#) et

[descendant](#)) de 134 Mbps qu'il faut ramener à 70 Mbps sur 20 km puisque les obstacles (maison, arbre, etc.) limitent la propagation du [signal](#).

Dans le cadre de la boucle locale radio, le WiMAX permet la fourniture d'un accès à [Internet](#) haut débit aux abonnés des zones non couvertes par les technologies filaires classiques de type [xDSL](#) ou Cable.

Le WiMAX permet également de relier entre eux des réseaux locaux sans fil de moindre portée tels que le [Wi-Fi](#) en remplacement des [backbones](#) filaires traditionnelles.

Retrouvez cette définition en ligne [[Lien22](#)]



Les derniers tutoriels et articles

Configurer un écran de démarrage graphique sous Debian

1. Introduction

Je remercie [gege2061](#) et [ovh](#) pour leur relecture de l'article et leurs conseils ;)



L'auteur ne pourra être tenu pour responsable en cas de problèmes lors du redémarrage de la machine. Si vous suivez à la lettre ce qui est dit dans ce tutoriel, vous n'aurez pas de problèmes.

Le boot splash permet d'avoir ceci avec la barre rouge qui avance avec le chargement de votre ordinateur.



Vous pourrez bien sûr changer le thème à votre guise. Pour avoir boot splash sous linux, il vous est nécessaire de patcher votre noyau. En effet, c'est ce dernier qui va faire la liaison entre le boot splash et l'affichage. Boot splash utilise le framebuffer. C'est une couche logicielle qui permet d'afficher des images en l'envoyant directement à la carte graphique. Voyons maintenant comment préparer le noyau. Ce tutoriel a été réalisé avec le noyau 2.6.16.5 sous Debian sarge.

2. Préparation et compilation du noyau

2.1. Préparation

Vous devez installer les sources de votre noyau. Si vous voulez avoir le noyau de votre debian, vous devrez taper cette commande en root

```
su
#saisie du mot de passe root
apt-get install kernel-source-2.6.8
```

Vous pouvez récupérer les sources d'un noyau tout neuf sur [kernel.org](#). Je vous conseille également le [ftp de free](#) qui est plus rapide que le site officiel.

Vous devez également récupérer un [patch](#) pour votre version de noyau. Celui utilisé dans ce cas est le [patch pour la version 2.6.15](#)



Le patch pour la version 2.6.15 fonctionne pour la version 2.6.16.5

Vous devez ensuite décompresser l'archive des sources du noyau

```
cd /usr/src
tar xvfj /home/user/linux-2.6.16.5.tar.bz2
```



Les fichiers d'extension .tar.bz2 prennent moins de place et donc moins de temps à télécharger

Ceci créera un dossier nommé linux-2.6.16.5 dans /usr/src. Vous devez maintenant patcher votre noyau. Cela veut dire que vous allez modifier son contenu en appelant un programme qui se charge de le faire pour vous.

```
cd /usr/src/linux-2.6.16.5/
patch -Np1 -i /home/user/boot splash-3.1.6-2.6.15.diff
```

Si tout se passe bien, vous ne devriez pas avoir de message d'erreur. Maintenant que le noyau est patché, il faut le configurer et le compiler. Vous avez plusieurs choix pour la configuration. Vous pouvez utiliser le mode console avec

```
Make autoconfig
```

ou le mode graphique. Pour lancer le mode graphique, vous devez installer gksu qui vous permettra d'accéder au mode graphique en root (nécessaire pour écrire le fichier de configuration du noyau). En utilisateur non privilégié, vous devrez lancer et entrer le mot de passe root le moment venu

```
# pour accéder à l'interface de configuration avec qt
gksu -g make xconfig
# ou pour accéder à l'interface de configuration avec
gtk
gksu -g make gconfig
```

Je ne détaillerais pas toute la configuration dans ce tutoriel mais seulement celle de boot splash. Une fenêtre comme celle-ci s'ouvrira pour configurer votre noyau.

Retrouvez la suite de l'article de Le_Coyote en ligne [[Lien21](#)]

Installation et gestion d'un UPS USB en réseau sous linux

Cet article présente l'installation d'un UPS (onduleur) de type USB de A à Z sur un serveur linux, ainsi que la gestion de l'UPS en réseau sur des machines linux et Windows.

1. Considérations générales sur les UPS

1.1. Définition et rôles

UPS signifie *Uninterruptable Power System*. Comme son nom l'indique, son rôle principal consiste à fournir une alimentation électrique continue quoiqu'il arrive. Plus précisément un UPS a 2 rôles :

- batterie de secours en cas de panne de courant : attention les batteries ne permettent que quelques dizaines de minutes d'autonomie (dépend de la charge à fournir). Un UPS a donc pour principal but de pouvoir éteindre proprement les machines qui lui sont connectées dès qu'une panne de courant survient, et d'éviter ainsi toute perte de données due à un crash disque provoqué par une interruption brutale d'alimentation.
- régulateur de tension : un bon UPS fournit un signal parfaitement sinusoïdal (stabilité de l'alimentation) et protège contre tous types de parasites, variations de tension et surcharges dues à la foudre, par exemple.

Outre son rôle purement électrique, l'UPS doit aussi communiquer avec l'ordinateur pour lui signaler son état et lui dire qu'il doit s'arrêter. Cette communication passe par un câble série RS-232, USB (de plus en plus fréquent), ou même par SNMP pour le haut de gamme. C'est cet aspect-là qu'il faut configurer et qui fait l'objet de cet article.

1.2. Types d'UPS

(souvent référencé sous le terme "topologie d'UPS") Il en existe 3 :

UPS offline : l'entrée de gamme, n'offre quasiment aucune régulation de tension, a pour seul but d'offrir une batterie de secours. Transmet intégralement toute perturbation électrique aux machines... A éviter autant que faire se peut, surtout pour des serveurs !

- **UPS "line interactive"** : milieu de gamme, offre une relative protection contre les surcharges avec un certain nombre de filtres électroniques. Si la foudre frappe violemment, la protection risque de ne pas être suffisante. Cependant dans la majeure partie des cas ce type d'UPS est satisfaisant.
- **UPS online** : le sommet de la technologie :) Régénère entièrement le signal (transformation alternatif --> continu --> alternatif), protection totale contre les surtensions. En cas de foudre c'est l'UPS qui encaissera (au point qu'il peut se détruire lui-même), mais les machines n'auront rien : la protection aura donc parfaitement rempli son devoir.

1.3. Comment choisir un UPS ?

Pour chacun des 3 types d'appareils, il existe plusieurs modèles, selon la puissance fournie, en VA (volt-ampère). Il est assez difficile d'estimer la puissance appropriée, mais il faut savoir qu'il

ne faut pas additionner la puissance brute des alimentations de PC (en simplifiant $W = VA$).

Par exemple un onduleur de 800 VA peut parfaitement protéger 3 PC normaux (sans raccorder les 3 écrans cependant) pendant 25 minutes (testé avec succès sur un UPS APC Back-UPS RS800). En effet un PC ne consomme jamais toute la puissance que son alimentation est capable de fournir. Naturellement, moins vous avez d'appareils connectés (graveur, lecteurs cd/dvd, disques etc.), moins de puissance vous consommerez.

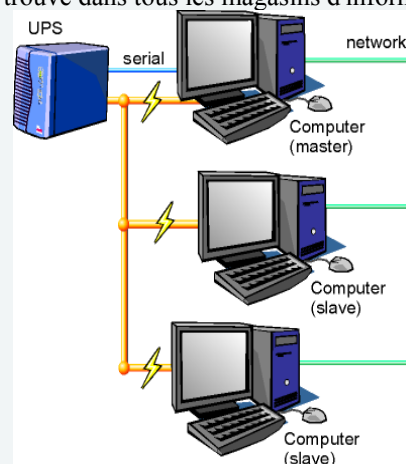
Pour plus de détails, reportez-vous au site suivant qui reprend pas mal de conseils : [\[Lien23\]](#)
[Calcul de puissance \[Lien24\]](#)

2. Application pratique

Notre cas de figure est une architecture de 3 serveurs : 2 linux [Debian\[Lien25\]](#) et 1 Windows 2000, tous 3 reliés à un switch Ethernet. L'UPS ne pouvant communiquer qu'avec une seule machine, on choisit un des serveurs linux comme étant le "maître", les 2 autres serveurs devenant les "esclaves". Les 3 serveurs devant échanger des données surtout lors d'une panne de courant, il est impératif que leurs alimentations soient toutes 3 branchées sur l'UPS, de même que le **switch réseau**, sinon toute communication sera impossible !

Dans notre cas, je dispose d'un UPS [APC Back-UPS RS de 800 VA\[Lien26\]](#), topologie *Line Interactive*.

L'UPS ayant des entrées tripôles type "PC", il est impossible d'y raccorder une fiche électrique européenne normale. Dès lors, pour y connecter le switch le mieux est de se servir d'un multi-prise spécial tel que le [MGE Pulsar CL5\[Lien27\]](#) qui présente l'avantage d'avoir un connecteur d'alimentation tripôle sous la forme "PC" standard. Il suffit dès lors de le relier à l'UPS par un câble d'alimentation mâle-femelle qu'on trouve dans tous les magasins d'informatique.



Voici le schéma de l'installation : But à atteindre : quand une coupure de courant survient (éventuellement après un certain délai si la panne n'est qu'une micro-coupure de quelques secondes), la machine maître doit envoyer un signal aux autres serveurs via le réseau pour leur commander une extinction automatique, puis le serveur principal doit s'éteindre à son tour.

Retrouvez la suite de l'article d'Olivier Van Hoof en ligne [\[Lien28\]](#)

Tests de la Mandriva 2006 PowerPack+

La distribution mandriva la plus complète est sans doute la distribution **PowerPack+**, elle intègre le système ainsi que toutes les fonctionnalités d'un serveur. Cette distribution est plus orientée pour les **PME-PMI**, offrant une solution de puissance et de simplicité.

1. Test de BWP-Necromance

1.1. Introduction

Mandriva, anciennement **Mandrake**, est la distribution linux française de référence. Depuis peu, cette distribution a fusionné avec 2 autres distributions (d'où le changement de nom): **Connectiva** et **Lycoris**, apportant des fonctionnalités en plus au système.

Dans cette article nous allons présenter ce système en détail et plus précisément la version PowerPack+ de **Mandriva**.

La boite PowerPack+ comprend :

- 2 DVD: Une version 32 Bits et 64 Bits
- 2 Manuels: Guide de démarrage et Guide d'administration Serveur (300 pages chacun)
- 7 CD avec 2 spécialement pour le serveur (Heartbeat et drdb)

1.2. Installation du système

L'installation du système ne change guère depuis l'époque de la mandrake. Elle est toujours aussi efficace et d'une simplicité déconcertante. Elle parcourt la plupart des principales tâches de l'installation:

- Le choix de la langue et du clavier
- La licence
- Choix du niveau de sécurité (Haute pour un serveur, normal pour un poste de travail)
- Le partitionnement du/des disque(s) installé dans la machine
- Une large sélection de sections de packages avec choix d'installation individuelle
- Le mot de passe ROOT
- La création d'utilisateur
- Gestion des périphériques (X, carte internet, firewall...)
- Mise à jour

L'installation dure environ 30 à 40 minutes selon la machine et selon les packages à installer. La mandriva installe beaucoup d'applications assez facilement ce qui permet d'installer un système stable et complet sans trop voir son fonctionnement - idéal pour les débutants qui veulent se mettre au monde de linux. Malheureusement, cette méthode peut ne pas plaire à des utilisateurs expérimentés qui préféreront une debian ou une slackware, voir une gentoo.

Néanmoins, la mandriva reste une bonne distribution et intègre urpmi, une application qui permet d'installer en ligne de commande des application sur le système avec ces dépendances avec.

Sans compter que cette version (comme la PowerPack) intègre directement les applications commerciales et les drivers officiels des cartes graphiques de type nvidia ou ati, ce qui est un plus et assez agréable.

J'ai été agréablement surpris puisque lors de ma configuration il était possible d'installer directement les drivers Windows de ma clé wifi USB et d'avoir le net installer par la suite sans faire 1h de configuration. De plus l'installation détecte automatiquement le matériel et s'adapte pour proposer une distribution pour la machine.

Le test a été porté sur une architecture 32 et 64 bit.

1.3. Les bureaux disponibles

1.3.1. GNOME 2.10



Le bureau GNOME

Le bureau GNOME en version 2.10 est sans aucun doute l'un des meilleurs bureaux qui existe sous Linux et en particulier sur cette distribution. Comme le montre le screenshot ci-dessus, on peut voir un bureau simple précis et agréable pour travailler. De plus, GNOME - plus axé coté entreprise - est nettement plus rapide que KDE. Et rajouter un wallpaper au couleur de la mandriva ne fait qu'améliorer encore plus le bureau.

1.3.2. KDE 3.4



Le bureau KDE 3.4

Le bureau KDE en version 3.4, moins bien que GNOME (avis personnel) ne s'en laisse pas défaire. Il intègre une multitude d'applications intéressantes et des thèmes qui font défaut sur GNOME. KDE est plus orienté pour les utilisateurs qui font un usage personnel. Toutefois, comme dit plus haut, je remarque une petite lenteur assez désagréable lors de chargement d'application comme firefox, kdevelop ... etc... mais n'est pas très important car l'utilisateur s'y fera sans problème.

1.3.3. IceWM

Le bureau IceWM ressemble à une interface de type Windows, pratique pour les personnes qui migrent de Windows vers Linux. Interface très sobre et rapide comme GNOME. Techniquement, c'est un bureau qui ne demande pas beaucoup de ressources: idéal pour les petites configurations .

1.3.4. WindowMaker

Le bureau WindowMaker est le bureau typique des système UNIX: simple et rapide. Sa grande valeur est qu'il marche pratiquement que par des terminaux.

1.3.5. BlackBox

BlackBox est comme FluxBox, un bureau très simple et qui demande peu de mémoire. Il est idéal pour les petites configurations ou pour faire une interface graphique pour un serveur afin de ne pas monopoliser les ressources de la machine.

1.4. Utilisation de la distribution

L'utilisation de la distribution PowerPack+ est très simple. Elle intègre une large palette d'outils pour permettre à l'utilisateur d'avoir un système complet installer sur son PC afin de pouvoir réaliser son travail. Parmi ces applications, on trouve les applications standards aux stations Linux/Unix comme gcc, apache2, samba, kdevelop...

Bien sûr, vous trouverez également des applications bureautiques comme OpenOffice 2, la suite gratuite. Avec cette suite, vous aurez un traitement de texte de bonne qualité compatible avec Word de Microsoft, un tableur compatible excel, un logiciel de dessin et un logiciel de base de données.

La distribution ne s'arrête pas à la bureautique et à des applications de type serveur, elle dispose de tous les outils pour graver des CD/DVD grâce à k3b, de lire des vidéos ou des mp3/ogg grâce à xine, totem ou xmms.

La distribution est bien fournie pour pouvoir l'utiliser correctement.

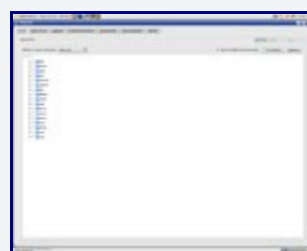
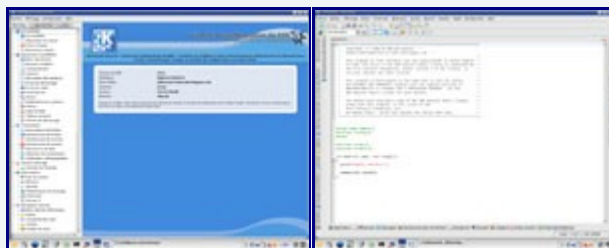
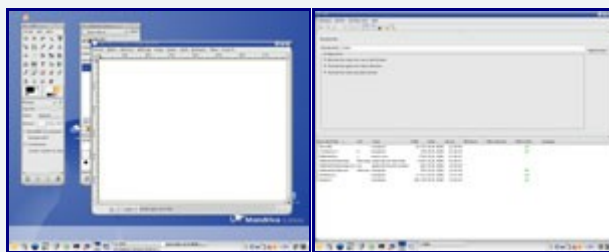
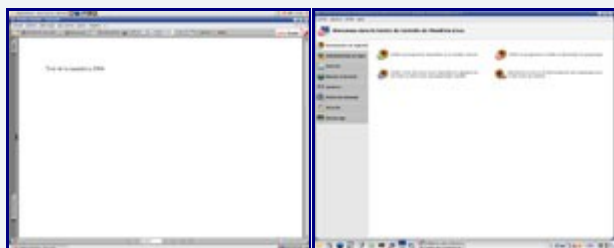
1.5. PowerPack ou PowerPack+ ?

Ceci est une bonne question, les 2 distributions de la Mandriva propose un système complet. La différence est que la distribution Powerpack+ propose 500 packages en plus destinés à l'installation d'un serveur complet.

Avec cette distribution, vous pouvez avoir le serveur Haute disponibilité HeartBeat et drdb.

Maintenant, vu le prix de commercialisation, si vous n'êtes pas une entreprise, la distribution PowerPack est largement suffisant pour des utilisateurs avancé sur linux.

1.6.Screenshots



1.7. Conclusion

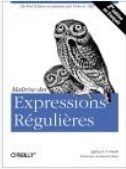
La conclusion de cette distribution est plutôt mitigée. Certes on a un système capable de faire beaucoup de choses, des outils propriétaires à souhait et une facilité déconcertante pour l'utilisation et la configuration. Mais après l'avoir utilisée, je remarque quelque "bugs" et une petite lenteur sur KDE, ce qui n'est pas trop gênant pour un utilisateur standard mais peut gêner un utilisateur avancé.

De ce fait, je vois mal cette distribution ce faire en tant que serveur, surtout où il faut de la stabilité.

Maintenant il est vrai que la Mandriva n'est pas mauvaise et loin de là. Elle est parfaite pour débuter dans le monde Linux en proposant divers outils de configuration comme DrakConf ou autres. On remarquera aussi l'ajout de la version 64bits dans la boîte ainsi que des outils pour serveur dans la version PowerPack+. L'installation quand à elle est très rapide et très simple et le matériel même récent est reconnu.

Donc si vous cherchez une distribution facile à utiliser, sûre au niveau de la sécurité et de ne pas passer des heures et des heures à installer et configurer, la Mandriva est la distribution qu'il vous faut.

Retrouvez la suite de l'article de 2Eurocents et BWP-Necromance en ligne [[Lien29](#)]



Maîtrise des expressions régulières

460 pages, 2^{ème} édition, mai 2003 Editions O'Reilly, ISBN: 2-84177-236-5
Public visé : **tous niveaux**

amazon.fr

Commandez sur Amazon.fr

Les expressions régulières sont de très puissants outils de manipulation de textes et de données. Elles ont fait florès ces dernières années au point de faire aujourd'hui partie intégrante de Perl, Java, VB.NET, C# (et tout langage basé sur la plate-forme .NET), PHP, Python, Ruby, Tcl, MySQL, awk, Emacs ainsi que de bien d'autres outils et langages grand public.

Grâce à cet ouvrage, le nouveau venu dans l'univers des expressions régulières découvrira une foule d'informations pour maîtriser ses données. L'utilisateur averti, de son côté, en appréciera la richesse et l'incomparable souci du détail. L'érudit, enfin, y trouvera matière à explorer de nouvelles pistes, loin des sentiers battus.



Linux en action

494 pages, 15 septembre 2005 Editions O'Reilly, ISBN: 2841773590
Public visé : **Initié**

amazon.fr

Commandez sur Amazon.fr

Linux en action rassemble une sélection de solutions efficaces aux problèmes le plus souvent rencontrés par les Linuxiens. À la fois source d'idées, moyen d'apprendre par l'exemple, cet ouvrage est construit sur le modèle "un problème / une solution / une discussion". Les recettes couvrent un éventail de tâches simples ou complexes. Cet ouvrage concerne principalement les distributions Red Hat, Fedora et Debian. Les débutants comme les plus avertis y trouveront des solutions rapides pour utiliser et administrer leur système Linux.

Critique du livre par la rédaction ([Nicolas Joseph](#)) :

Ce livre s'adresse aux personnes ayant installés Linux sur leur ordinateur (donc ne cherchez pas de chapitre Installation) et qui souhaitent maîtriser ou tout simplement comprendre le fonctionnement de leur système. Ce livre s'adresse aussi aux administrateurs novices qui y trouveront la démarche à suivre pour configurer correctement leurs machines. On retrouve le découpage de chaque partie en problème/solution/discussion/voir aussi, qui m'avait déjà séduit dans [http://c.developpez.com/...\[Lien30\]](http://c.developpez.com/...[Lien30]), qui en fait un livre agréable à lire et très pratique pour trouver des solutions. Certains points abordés sont spécifiques à chaque distribution (tel que la gestion des paquets). Dans ce cas, il est traité une partie pour la distribution Debian et une autre pour la RedHat et, si elle existe, la méthode générale (la compilation des sources dans le cas de la gestions des paquets). La seule réelle déception que j'ai eu à la lecture de ce livre a été la pauvreté des codes proposés! En fait il ne s'agit que de suite de commandes à taper dans un terminal. Il aurait été très intéressant de multiplier les exemples tel que l'ajout des utilisateurs par lots ou le changements

Critique du livre par la rédaction ([Laurent Dardenne](#)) :

Un des meilleurs livres d'informatique que j'ai pu lire ces derniers temps ! L'auteur fait preuve d'une pédagogie remarquable qu'on soit débutant ou non sur le sujet. L'approche des expressions régulières débute très simplement puis au fil des pages on entrevoit toute la puissance de cet outil de manipulation de texte et de données mais aussi toutes ses chausse-trappes.

Pour les utilisateurs avertis cet ouvrage de référence est à placer bien en évidence sur le bureau tant les informations sont nombreuses et difficile à mémoriser après une première lecture.

Il est préférable pour les débutants que sa lecture se fasse devant un ordinateur afin de pratiquer rapidement les exercices proposés.

Le langage privilégié est PERL bien que les moteurs les plus courant soit abordés : .NET (C# et VB.NET notamment), Java, Python, Tcl, ... Quoi dire de plus sinon que tout développeur devrait avoir cet ouvrage dans sa bibliothèque et que sa lecture modifiera certainement votre manière d'aborder les problèmes de manipulations de texte. Bonne lecture.

des mots de passe par lots. A l'exception de ce dernier point, *Linux en action* est un bon livre pour commencer à maîtriser Linux et ensuite pouvoir aborder des livres plus complexes.

Critique du livre par la rédaction ([Matthieu Petiot](#)) :

Ce livre s'adresse à toutes les personnes qui possèdent un système Linux à la maison ou sur leur lieu de travail et qui souhaitent l'administrer simplement mais avec efficacité. C'est ainsi que les problèmes les plus courants y sont répertoriés. L'auteur explique, au cours de plusieurs chapitres, comment gérer les paquets en fonction de la distribution utilisée : Rpm, Debian ou compilation des sources. Les autres gestionnaires de paquets ne sont pas présentés, mais il y en a tellement... Les connaisseurs trouveront sûrement ces parties superflues mais elles seront bien utiles pour les personnes qui débutent.

Ce livre apporte les recettes déjà cuisinées pour mettre en place un serveur web Apache, mettre en place le partage de fichiers grâce à SaMBA, configurer son imprimante grâce à CUPS, configurer un serveur de mail ou encore paramétrer un serveur DHCP. Il s'agit des problèmes que rencontreront un jour ou l'autre la plupart des linuxiens. Qui n'a jamais été confronté à un problème matériel avec Linux ? Pas moi en tout cas... Ce livre tente donc d'apporter un support de réponse à ces problèmes.

J'ai bien apprécié la partie sur la gestion des groupes, des utilisateurs et des droits, un rappel complet ne fait jamais de mal. Je regrette toutefois que certaines parties ne soient que peu approfondies. Ce qui est notamment le cas pour la partie concernant les clés publiques et privées de SSH ou encore la mise en oeuvre d'un serveur de temps NTP.

J'ai trouvé ce livre fort intéressant mais quelques fois pauvre au niveau des informations fournies. Cependant, je pense qu'il répondra largement aux attentes de débutants qui ne souhaitent pas forcément configurer un logiciel de A à Z.



Prise de contrôle à distance sur un PC non-accessible

Bonjour,

Le but de ce post est de montrer comment accéder à une machine A depuis une autre B via SSH. A pouvant se connecter à Internet mais B ne pouvant pas voir A (je ne sais pas si je suis très claire 😊)

En fait A est le client chez qui je veux me connecter pour la maintenance. B est mon ordinateur. A peut pinguer mon IP mais B ne peut pas pinguer A.

Chaque PC doit avoir un serveur SSH.

Ainsi, A doit se connecter sur B via la commande :

```
ssh -R2222:localhost:22 -l user_chez_B  
IP_de_B
```

A est donc connecté à B.

Maintenant B peut se connecter à A de cette manière :

```
ssh -p2222 -l user_chez_A 127.0.0.1
```

Je ne suis pas expert en SSH mais je trouve cette fonctionnalité très pratique pour effectuer la maintenance chez les clients sans modifier leur configuration réseau.

Après il est très simple de faire un petit script shell qui va demander l'IP de B un login et un mot de passe, lancer le serveur SSH et lancer la connexion sur B 😊

Retrouvez le blog de narmataru en ligne [[Lien31](#)]

Vu sur Developpez.tv

Retrouvez les vidéos des conférences des **Solutions Linux 2006**.

Événement filmé par [Sylvain Luce](#). Vidéos au format XVID et Windows Media Video.

Bon visionnage !

Visionnez ces reportages en ligne [[Lien32](#)]



Delphi



Les derniers tutoriels et articles

Migration de code Delphi Win32 vers .NET – Partie 1

Si le développement de nouvelles applications sous .NET ne demandent qu'à suivre la logique du langage et de la bibliothèque d'affichage choisis, reprendre un code VCL Win32 pour en assurer la migration sous VCL.NET peut présenter des écueils malgré l'effort important que Borland a fourni pour assurer une compatibilité ascendante proche de la perfection. Savoir comment procéder est ici tout aussi essentiel que de savoir où se situeront les zones délicates pour les isoler rapidement. Le présent article ainsi que le suivant vous montreront à la fois la méthode à utiliser et les pièges à éviter pour réussir vos migrations.

Le portage d'une application d'une plate-forme à une autre n'est jamais dépourvu de surprises, bonnes et mauvaises, qui rendent souvent la tâche ardue et hasardeuse en terme de budget (temps et financier). Savoir ce qu'il faut migrer et comment effectuer cette migration est d'une importance au moins aussi grande que de savoir comment, techniquement, coder ou re-coder telle ou telle partie d'une application.

1. L'intérêt de passer en VCL.NET

La question mérite d'être posée et cette section sur le portage des applications semble un bon endroit pour y répondre... En effet, pour une application existante, vaut-il mieux la passer en VCL.NET ou bien la refaire en Windows Forms ? En dehors du gain de temps évident qui prêche en faveur de la VCL.NET, le débat reste ouvert. Un développeur Delphi qui utilisera Delphi.NET et la VCL.NET ne perdra aucun temps en formation lourde sur un nouveau langage et une nouvelle bibliothèque d'affichage et cela est essentiel. D'un point de vue commercial, il est toujours préférable de sortir un produit perfectible en temps et en heure plutôt qu'une perfection (jamais atteinte) mais après la concurrence...

Si on met de côté ces aspects liés au temps d'apprentissage, le choix de la VCL.NET n'est pas « pire » techniquement que Windows Forms, en tout cas en première approximation. En effet, le futur de l'affichage sous .NET est XAML et Windows Vista sort en 2006. A ce moment là Windows Forms sera-t-elle une librairie d'affichage « has been » ? Cela est difficile à dire mais tout porte à croire qu'à terme XAML remplacera totalement Windows Forms. Cette dernière aura peut-être encore un avenir grâce aux unités mobiles et au Compact Framework ou bien grâce à son portage sous MONO.

Dès lors choisir VCL.NET est tout aussi « risqué » que de choisir Windows Forms. Reste la cohérence des développements. En effet,

VCL.NET n'est utilisable que sous Delphi.NET alors que Windows Forms est utilisable par tous les langages de la plate-forme, dont Delphi.NET. De fait, celui qui doit créer des composants visuels aur-t-il intérêt à le faire sous WF plutôt que sous VCL.NET afin d'élargir considérablement sa base d'utilisateurs potentiels sans se priver des utilisateurs Delphi.

Celui qui doit aussi intégrer des travaux venant d'autres développeurs ou qui doit créer des assemblages utilisables dans d'autres langages aura-t-il tout intérêt à choisir Windows Forms plutôt que VCL.NET. Le choix Windows Forms ou VCL.NET n'est pas sans conséquence et nous vous laissons tirer vos conclusions de ces réflexions et choisir la meilleure solution pour vos développements.

Accéder à une base de données ODBC avec une édition Personnelle de Delphi.

Ce tutoriel propose aux possesseurs de Delphi Edition Personnelle l'accès aux données d'une base relationnelle via les API ODBC. Pour ce faire, nous utiliserons une traduction en Delphi des librairies sql.h livrée par Microsoft.

1. Informations Générales

1.1. Introduction

Depuis Delphi 6, Borland met à disposition des versions gratuites de son outil de développement "Delphi". Ces versions sont nommées "éditions personnelles". Cela permet de réaliser et de distribuer, dans le cadre stipulé dans la licence d'utilisation des éditions personnelles, des applications sans avoir à investir dans un outil.

Même si ces versions sont gratuites, elles contiennent suffisamment pour réaliser un grand nombre d'applications. La partie faisant souvent défaut est l'accès aux bases de données car aucun composant n'est fourni dans une édition personnelle.

Ce manque peut être comblé de diverses manières, par l'ajout de composants tiers, par utilisation d'API fournies avec la base de données ou bien alors par connexion ODBC.

Pour les possesseurs des versions 6 et 7 de Delphi, vous pouvez également accéder à des bases de données en utilisant les bibliothèques de types ADO. Pour cette technique, vous pouvez vous reporter à [l'article\[Lien35\]](#) écrit par Thierry AIM. Cette technique n'est malheureusement pas applicable à Delphi 2005 car il ne permet malheureusement pas l'importation de bibliothèques de types.

Nous allons donc voir la manière d'utiliser Delphi pour accéder aux bases de données via les fonctions API de l'ODBC.

Une fois ce tutoriel parcouru, vous pourrez consulter la [documentation des fonctions ODBC\[Lien36\]](#) pour une utilisation plus approfondie.

1.2. Qu'est-ce que l'ODBC ?

L'ODBC (**O**pen **D**ata**B**ase **C**onnectivity) est une couche logicielle (*Middleware*) de Windows permettant à un programme d'accéder à des bases de données. Pour que cela fonctionne, la base de données doit fournir un pilote compatible ODBC afin de traduire les ordres SQL du programme en ordres compréhensibles par la base de

2. Portage de code

Toutes fonctions, bibliothèques de fonctions et autres codes, qui ne font aucun appel à la plate-forme Windows ni à l'interface sont portables dans les meilleures conditions qu'on puisse imaginer. Toutefois, il convient de bien faire attention à la nouvelle donnée introduite par le passage à .NET. Toutes les nouveautés que nous avons vues jusqu'ici à propos du langage s'appliquent bien entendu à tout code migré de Win32 à .NET autant qu'au code créé *ex nihilo*.

Retrouvez la suite de l'article d'Olivier Dahan en ligne [\[Lien33\]](#)

Retrouvez la seconde partie de l'article en ligne [\[Lien34\]](#)

données utilisée.

Les postes Windows disposent souvent d'un minimum de pilotes fournis par Microsoft qui contiennent entre autre un pilote pour Access, permettant ainsi d'interagir avec ce type de base sans pour autant disposer du logiciel Access complet.

Ces pilotes sont distribués au travers du MDAC, qui est livré en standard avec Windows XP. Pour télécharger les dernières versions, direction le [site de Microsoft\[Lien37\]](#).

1.3. Documentation ODBC

Pour en savoir plus, vous pouvez consulter la documentation anglaise fournie par Microsoft dans son [MSDN\[Lien38\]](#).

La liste et descriptions, en anglais, des fonctions ODBC sont disponibles [ici\[Lien39\]](#).

1.4. Interface ODBC utilisée

Pour pouvoir utiliser les exemples qui seront donnés par la suite, vous devez utiliser une unité Delphi contenant toutes les définitions des fonctions et types utilisés par l'interface ODBC.

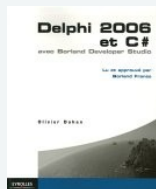
Pour cela, téléchargez l'archive [ZIP suivante\[Lien40\]](#) qui contient l'unité "iodbc.pas" que vous n'aurez qu'à inclure dans la clause uses de votre projet.

Cette unité a été réalisée en se basant sur la version 3.51 de l'ODBC.

2. Paramétrage de la source ODBC

Il existe 2 méthodes pour se connecter à une source ODBC. La première à l'aide du gestionnaire de sources ODBC de Windows (manuellement ou par programmation), la seconde en indiquant le nom et l'emplacement de la base de données.

Retrouvez la suite de l'article de Michaël Duval en ligne [\[Lien43\]](#)



[Livre "Delphi 2006 et C#" par Olivier Dahan de la rédaction de www.developpez.com \[Lien45\]](#)

[Olivier Dahan \(Merlin\)](#) est un des membres fondateurs de www.developpez.com, et participe depuis sa création fin 1999 à l'équipe de rédaction Delphi. Il est déjà l'auteur de nombreux livres à succès sur Delphi. Son dernier livre est désormais disponible :

Delphi 2006 et C# : Avec Borland Developer Studio

Le savoir-faire d'un expert français de Delphi et C#. Approuvé par Borland France et écrit par l'un des principaux acteurs de la scène Borland en France, cet ouvrage offre une expertise unique sur la programmation sous .NET et C#. De la programmation des

Windows Forms à la conception de sites Web sous ASP.NET en passant par l'accès aux données avec ADO.NET et l'écriture de composants .NET, il permet aux développeurs Delphi d'acquérir une double compétence C# et Delphi, ouvrant ainsi des perspectives vers ce que sera le développement multi-plate-forme dans les années à venir. A qui s'adresse cet ouvrage ? - Aux développeurs Delphi souhaitant migrer vers Delphi.NET ; - Aux développeurs Delphi souhaitant s'initier à C# en douceur ; - Aux étudiants en informatique désirent s'initier à .NET ainsi qu'à tous ceux que .NET interpelle et qui désirent comprendre cette plate-forme sans se limiter à un seul langage de programmation.

[Commandez sur Amazon.fr \[Lien45\]](#)- Livraison gratuite

Tutoriel Delphi : Introduction aux RTTI

L'objectif premier des informations de types à l'exécution (alias RTTI) est de permettre à l'IDE de reconnaître et de manipuler des classes, plus précisément des composants, qui ont été définis bien après la commercialisation de Delphi.

Le mode conception de fiche ne serait pas possible sans le mécanisme de RTTI qui est également lié à la sérialisation, c'est à dire la persistance sur disque des fiches (forms) conçues et

contenant des composants (fichiers .DFM). Je n'aborderai pas ici la sérialisation, le sujet nécessitant à lui seul un tutoriel.

RTTI est donc la clef de voûte entre les composants Delphi et leur incorporation et manipulation dans L'EDI. Il permet de retrouver des informations sur le type d'une instance de classe lors de l'exécution d'un programme, ce type pouvant être inconnu lors de la compilation.

[La suite du tutoriel\[Lien46\]](#)

News

Téléchargez : la documentation de Delphi 7 en Français



1. Prise en main

Le manuel Prise en main présente l'environnement de développement de Delphi 7 pour vous permettre d'utiliser immédiatement le produit.

Il indique aussi où rechercher des renseignements sur l'utilisation des outils et des fonctionnalités disponibles dans Delphi 7.

56 pages - 928 Ko - [Mirroir1](#) [Miroir2](#)

2. Guide du développeur

Le Guide du développeur aborde des notions de développement intermédiaires et avancées.

Il traite notamment de la création d'applications de bases de données client/serveur, de la création d'applications serveurs Web Internet et de l'écriture de composants personnalisés. Il vous permet de construire des applications qui respectent les spécifications de nombreux standards comme SOAP, TCP/IP, COM+ et ActiveX. Nombre de fonctionnalités avancées concernant le développement web, les technologies XML de pointe et le développement de bases de données nécessitent des composants ou des experts qui ne sont pas disponibles dans toutes les éditions de Delphi.

Le Guide du développeur suppose que l'utilisation et les techniques fondamentales de programmation Delphi ont été assimilées.

Pour une présentation de la programmation Delphi 7 et de l'environnement de développement intégré (EDI), voir [le manuel de Prise en main](#) et l'aide en ligne (F1).

1186 pages - 14,85 Mo - [Mirroir1](#) [Mirroir2](#)

Retrouvez les documentations en ligne [\[Lien47\]](#)

Débat : Que vous inspire la décision de Borland de vendre ses EDI ?

Venez participer sur les forums au débat sur la vente des EDI par Borland [\[Lien48\]](#)

Web 2.0, allez plus loin avec AJAX et XMLHttpRequest

Si il y a un terme à la mode en ce moment c'est bien celui d'AJAX, on le retrouve un peu a toutes les sauces. Qu'est ce donc que cet AJAX ? Nous verrons à quoi correspondent ces lettres mais surtout les concepts qui se cachent derrière et les différentes utilisations que l'on peut en faire.

1. Introduction

Qu'est ce qu'AJAX ?

Vous trouverez certainement plusieurs définitions différentes à cet acronyme.

En voici une : **Asynchronous JavaScript And XML**.

AJAX n'est donc pas une technologie mais plutôt un concept qui permet donc de faire des appels asynchrones au serveur depuis le client.

Lors de ces appels, le serveur retournera du XML qui sera "récupéré" par javascript et traité.

Nous verrons que nous pouvons tout aussi bien faire transiter du texte et faire des appels synchrone si l'on veut.

Avant toute chose, il serait bon de faire un point sur le processus classique de consultation d'un site ou d'une application web :

1. Vous saisissez une adresse dans votre navigateur.
2. Cette "requête" finie par arriver sur le serveur web qui héberge la page en question.
3. Le serveur vous retourne du texte au format HTML ou XHTML et éventuellement des images, feuilles de style, fichiers JavaScript, applets java
4. Votre navigateur les interprète et vous affiche la page.
5. Vous êtes déconnecté du serveur web.

Donc, quand vous cliquez sur un lien, vous recommencez ce processus en entier avec une nouvelle page.

Dans le cas ou un formulaire se trouve sur la page, vous envoyez les données sur le serveur qui vous répondra après traitement de ces données.

L'utilisation d'AJAX va chambouler un peu cette organisation car a tout moment vous pouvez aller chercher des informations sur le serveur pour :

- Ajouter des éléments a la page
- Modifier le contenu d'un "bout de la page"
- Insérer des données dans une base.

Les applications sont donc très nombreuses :

- Champs qui s'auto-complètent (comme google suggest).
- Listes déroulantes liées.
- Contrôle de formulaire.

Ajax - une autocomplétion pas à pas

Ajax est un terme très à la mode actuellement dans les développements web. Cet article propose de créer pas à pas un script d'auto-complétion utilisant cette technologie, dans deux buts : créer une interface conviviale d'aide à la saisie (type Google Suggest) mais aussi et surtout voir ce qui se cache vraiment sous la dénomination d'Ajax.

1. Introduction

Ajax est un acronyme très à la mode de nos jours. Qui n'a pas rêvé de pouvoir rajouter le prestigieux label Ajax (Asynchronimous Javascript And Xml) à un de ses sites web? Le but de cet article est de reprendre pas à pas le mécanisme d'auto-complétion utilisant la technologie Ajax mis en place à de nombreux endroits. L'exemple

- Popups accessibles.
- ...

Mais comment faire pour l'utiliser ?

2. L'objet XMLHttpRequest

AJAX se base sur l'utilisation d'un composant embarqué dans presque tous les navigateurs récents.

Par contre, vous vous doutez bien que le comportement va varier en fonction de ces derniers.

Pour pouvoir utiliser AJAX, il nous faut donc créer en javascript un objet que l'on nomme **XmlHttpRequest** ou **xhr** pour les intimes, comme son nom l'indique, il va nous permettre de faire des requêtes http pour échanger du XML.

Création de l'objet XMLHttpRequest


```
var xhr = null;

if(window.XMLHttpRequest) // Firefox et autres
    xhr = new XMLHttpRequest();
else if(window.ActiveXObject){ // Internet Explorer
    try {
        xhr = new
ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        xhr = new
ActiveXObject("Microsoft.XMLHTTP");
    }
} else { // XMLHttpRequest non supporté par le
navigateur
    alert("Votre navigateur ne supporte pas les
objets XMLHttpRequest...");
    xhr = false;
}
```

Comme vous pouvez le constater, Internet Explorer se démarque des autres.

A son crédit on peut quand même préciser que rien n'est standardisé à ce niveau là et que même si c'est un ActiveX, il ne fait pas apparaître de message de sécurité.

Retrouvez la suite de l'article de François Dussert en ligne [[Lien49](#)]

le plus parlant, et peut-être le plus abouti est celui de  [Google Suggest\[Lien50\]](#) qui vous propose au fur et à mesure que vous entrez le texte de votre recherche les complétions les plus populaires.

Pour ce genre de choses, il est bien entendu impensable de passer

dès le chargement de la page web l'ensemble des informations aux navigateurs clients. Le volume que cela représenterait saturerait le serveur en un temps record! C'est là qu'intervient notre fameuse fée magique Ajax, qui va aller chercher l'information sur le serveur et l'intégrer au client sans que celui-ci ait à subir un rechargement de complet de la page.

Pour cet exemple, nous utiliserons un script très basique pour le serveur de données (une simple page PHP) car ce qui nous intéresse le plus est bien évidemment la mise en place de tout le moteur JavaScript nécessaire pour faire fonctionner notre auto-complétion. De plus, la compatibilité de Google Suggest avec les navigateurs anciens est très poussée. Dans le but de rendre les scripts plus lisibles, nous nous contenterons d'une version qui fonctionne avec les navigateurs récents (IE 6, Firefox 1.5 et partiellement Opera 8.5).

Nous allons créer un script permettant de gérer une liste déroulante, dans le sens habituel du terme, qui présentera les fonctionnalités qu'un utilisateur est en droit d'attendre d'une telle liste : proposition de complétion du champ texte, surlignement de la suggestion active, navigation par les flèches haut/bas ou à la souris, ...

Dans un souci de clarté, nous mettrons en oeuvre les fonctionnalités de notre script d'auto-complétion progressivement, en trois grandes étapes: La mise en place du dialogue client/serveur, la mise en place de la présentation et enfin la gestion des évènements.

2. Etape 1 - Faire communiquer le client et le serveur

Cette première étape est celle où nous allons mettre en oeuvre le fameux objet XMLHttpRequest qui va nous permettre de faire communiquer notre navigateur web avec le serveur, sans que l'utilisateur ait besoin de recharger sa page.

C'est le coeur de toute page utilisant Ajax, et la nouveauté introduite par cette technologie. Mais pour autant, ce n'est absolument pas la partie la plus difficile de la chose...

L'objet XMLHttpRequest permet comme son nom l'indique d'effectuer une requête HTTP vers notre serveur (et uniquement celui-là pour des raisons de sécurité), et d'effectuer un traitement dans notre page au moment du retour de la requête. Dans notre cas, la requête nous donnera les 10 premières possibilités de complétion de notre champ texte.

2.1. Côté serveur

Tout d'abord intéressons-nous à la page coté serveur qui va renvoyer les possibilités de complétion à notre utilisateur. Comme il ne s'agit pas vraiment du sujet de cet article, nous allons la réduire à son strict minimum de façon à ce qu'elle nous retourne, pour une entrée utilisateur donnée, un fichier XML contenant les complétions possibles.

Voici le fichier php que nous utiliserons :

```
<?php
header('Content-Type: text/xml;charset=utf-8');
echo(utf8_encode("<?xml version='1.0' encoding='UTF-8' ?
><options>"));
if (isset($_GET['debut'])) {
    $debut = utf8_decode($_GET['debut']);
} else {
    $debut = "";
}
$debut = strtolower($debut);
$liste = array([...]);
function generateOptions($debut,$liste) {
    $MAX_RETURN = 10;
    $i = 0;
    foreach ($liste as $element) {
```

```
        if ($i<$MAX_RETURN && substr($element, 0,
strlen($debut))== $debut) {
            echo(utf8_encode("<option>".$element."</opti
on>"));
            $i++;
        }
    }
}
generateOptions($debut,$liste);
echo("</options>");
?>
```

La liste de mots utilisée (dans la variable \$liste, qui n'a pas été recopiée ici) est la liste des 1500 mots maîtrisés par les élèves de CE2.

Vous pouvez tester le retour de cette page php: [Liste des mots commençant par "de"](#) [Lien51].

Le nombre maximum de possibilité retournée par cette page est 10. Dans une application réelle, on peut tout à fait envisager que les complétions les plus probables soient retournée en premier (comme pour Google Suggest). Ces traitements sont à la charge du serveur et à votre initiative.

Le seul point particulier de cette page php est de bien noter que, de façon générale, il convient d'envoyer la réponse XML en UTF-8 bien propre, pour éviter d'éventuels problèmes d'encodage par après (accents qui disparaissent, document XML non reconnu...).

2.2. Côté client

2.2.1. La page HTML

Côté client, nous allons commencer par mettre en place une page HTML, la plus simple possible. Elle sera complétée par deux balises script qui respectivement la liera à notre script javascript et initialisera ce script.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr"
lang="fr">
<head>
<title>Test d'autocomplétion</title>
</head>
<body>
<form name="form-test" id="form-test"
action="javascript:alert('soumission
de ' + document.getElementById('champ-texte').value)"
style="margin-left: 50px; margin-
top:20px">
<input type="text" name="champ-texte"
id="champ-texte" size="20" />
<input type="submit" id="bouton-submit">
</form>
</body>
</html>
```

Le body de la page ne changera pas de tout l'article. Nous nous contenterons d'y connecter les scripts nécessaires. Ainsi, pour un utilisateur n'ayant pas activé Javascript (un peu près 10% des internautes), le formulaire apparaîtra comme un formulaire normal, sans aide à la complétion.

2.2.2. L'objet XMLHttpRequest

Le premier script que nous allons mettre en place est celui permettant de créer un objet XMLHttpRequest (ou XHR pour les intimes). Cet objet va nous permettre d'effectuer des requêtes vers notre serveur, sans avoir à recharger entièrement la page. Pour plus d'informations sur l'objet et ces méthodes, consultez [l'article de](#)

[siddh sur le sujet\[Lien52\]](#).

Pour notre part, nous allons utiliser la méthode suivante permettant de créer un nouvel objet, compatible entre tous les navigateurs actuels supportant l'objet XMLHttpRequest:

```
// retourne un objet XMLHttpRequest.  
// méthode compatible entre tous les navigateurs  
(IE/Firefox/Opera)  
function getXMLHTTP(){  
    var xhr=null;  
    if(window.XMLHttpRequest) // Firefox et autres  
        xhr = new XMLHttpRequest();  
    else if(window.ActiveXObject){ // Internet Explorer  
        try {  
            xhr = new ActiveXObject("Msxml2.XMLHTTP");  
        } catch (e) {  
            try {  
                xhr = new ActiveXObject("Microsoft.XMLHTTP");  
            } catch (e1) {  
                xhr = null;  
            }  
        }  
    }  
    else { // XMLHttpRequest non supporté par le  
navigateur  
        alert("Votre navigateur ne supporte pas les objets  
XMLHttpRequest...");  
    }  
    return xhr;  
}
```

2.2.3. Mise en place des constantes

Pour fonctionner et se mettre en place, notre script va avoir besoin de 3 constantes:

Vue d'ensemble de l'ergonomie sur Internet - Bonnes Pratiques

Pourquoi faire un article sur les bonnes pratiques d'ergonomie? Et bien je me suis rendu compte, au travers des divers projets sur lesquels j'ai travaillé, que l'aspect ergonomique des applications était, dans la majeure partie des cas, laissé à la charge du développeur. La personne en charge de cette tâche est généralement plus habituée à réaliser de beaux algorithmes, plutôt que de beaux écrans. Ce document donne un panel d'astuces permettant la réalisation de sites plus ergonomiques et attractifs.

1. Avant-Propos

Il existe différents types de sites Internet : site de e-commerce, intranet de société, site personnel.

Tous ont pour objectif, pour des raisons différentes, d'accroître le nombre de visites, et de fidéliser un grand nombre d'internautes. Ce document donne une vue d'ensemble sur les bonnes pratiques à mettre en oeuvre en terme d'ergonomie sur une application Internet. Le terme « ergonomie » englobe les notions de « design » et d'« usabilité ». La satisfaction des utilisateurs est une des notions fondamentales à prendre en compte pour la bonne réalisation d'un site web.

« La conception d'un site utile et performant demande non seulement que les contenus et services répondent aux besoins des utilisateurs, mais surtout que l'ergonomie et la navigation en soient pensées afin de les rendre facilement accessibles aux utilisateurs finaux. »

Les éléments énumérés dans ce document ne constituent pas une liste exhaustive, mais un aperçu des éléments à prendre en compte.

- `_documentForm` : le formulaire contenant notre champ texte
- `_inputField` : le champ texte lui-même
- `_submitButton` : le bouton submit de notre formulaire

La fonction d'initialisation de notre script prendra donc comme argument une référence sur ces trois éléments.

```
var _documentForm=null; // le formulaire contenant notre  
champ texte  
var _inputField=null; // le champ texte lui-même  
var _submitButton=null; // le bouton submit de notre  
formulaire  
function initAutoComplete(form,field,submit){  
    _documentForm=form;  
    _inputField=field;  
    _submitButton=submit;  
    _inputField.autocomplete="off";  
}
```

On désactive l'aide à la saisie des navigateurs en mettant l'attribut autocomplete de notre champ texte à *false*.

Cette fonction sera liée à l'évènement window.onload de notre page. Elle sera enrichie au fur et à mesure par des besoins supplémentaires d'initialisation que nous rencontrerons tout au long de cet article.

```
<script type="text/javascript">  
window.onload =  
function(){initAutoComplete(document.getElementById('form-  
m-test'),  
    document.getElementById('champ-  
texte'),document.getElementById('bouton-submit'))};  
</script>
```

Retrouvez la suite de l'article de Denis Cabasson en ligne [\[Lien53\]](#)

2. Généralités sur l'ergonomie

2.1. Axes de travail

Un site web a pour objectif de fidéliser ses utilisateurs présents et à venir. Il existe 3 axes généraux permettant la réalisation de cet objectif :

Un contenu cohérent en corrélation avec les attentes de l'utilisateur. Une architecture et une navigation performantes et facilement compréhensibles. Un environnement graphique agréable et structuré.

Les 5 facteurs de succès d'un site web selon Jakob Nielsen* (1)

- **La facilité d'apprentissage et de compréhension.** Tout utilisateur qui n'a jamais vu le site doit en comprendre rapidement le fonctionnement.
- **L'efficacité d'utilisation.** L'utilisateur doit trouver rapidement l'information qu'il cherche ou réussir à acheter rapidement le produit voulu.
- **La facilité de mémorisation.** Les utilisateurs, même après une période de non utilisation, doivent pouvoir utiliser le site sans avoir à réapprendre son fonctionnement.
- **L'utilisation sans erreur.** L'utilisateur doit atteindre l'objectif visé sans effectuer de « fausses manoeuvres ».
- **La satisfaction.** C'est l'appréciation subjective du site par l'utilisateur, conditionnée par les aspects esthétiques du

site, ainsi que par les 4 attributs précédents.



Un moyen efficace de vérifier si la mise en oeuvre de ces 3 critères est réussie est : Le test utilisateur. Au travers de ces tests, l'utilisateur fera remonter des informations précises sur les défauts de navigation, de cohérence, et de structure du site, autrement dit d'ergonomie. La précision de ces informations optimisera les délais d'amélioration des points noirs de l'environnement.

2.2. L'utilisateur

La lecture d'une page par un utilisateur représente une certaine charge de travail. : appréhension des informations, leur compréhension, décisions à prendre, etc
i cette charge devient trop importante elle risque de le décourager et de diminuer fortement son intérêt pour le site qu'il est en train de visiter.

L'internaute qui voit une page pour la première fois, commence par la scanner. C'est-à-dire que son oeil parcourt rapidement l'ensemble de l'écran afin d'en avoir un aperçu global. C'est à ce moment crucial que le lecteur va se faire une idée du contenu qu'il a sous les yeux.



L'être humain possède essentiellement 2 types de mémoire. Une à long terme (mémoire de sa vie) et une autre à court terme (ou mémoire de travail).

Lors de la première lecture (sans répétition), c'est la mémoire de travail qui est sollicitée. Elle peut, selon les personnes, stocker environs 7 éléments (+/- 2).

Quelques règles basiques régissent la perception visuelle d'un internaute :

- De manière générale, la vision regroupe instinctivement les objets vus selon leur similitude graphique.
- Leur proximité relative .
- Et leur régularité spatiale
- Pour leur donner la même valeur sémantique.

Tous ces éléments sont à prendre en compte lors de la conception ergonomique d'un site internet.

Retrouvez la suite de l'article de Fabrice Sznajderman en ligne [[Lien54](#)]

Vu sur les Blogs

Navigateur et standards Web !

Il est toujours bon d'avoir un aperçu du support des standards du Web par les principaux navigateurs du marché.

Le site [Web Devout](#) propose justement un petit récapitulatif du support des principaux standards du Web (HTML/XHTML, CSS 2.1 et 3, DOM, et ECMAScript, la version normalisé de JavaScript).

J'ai surtout été étonné de voir que l'**ECMAScript** se plaçait en tête, et qu'il est même complètement supporté par les principaux navigateurs (le seul problème, plutôt mineur, vient de la méthode **split()** sous Internet Explorer qui n'est pas tout à fait conforme).

En fait les principaux problèmes du **JavaScript** proviennent soit du modèle objet (**DOM**), soit de fonctionnalités supplémentaires ajoutés par les différents navigateurs...

On s'aperçoit également que l'écart s'est creusé entre Internet Explorer et ses alternatives concurrents. A sa sortie Internet Explorer 6 était un des navigateurs le plus conforme... Aujourd'hui

Comment installer RoR manuellement ?

Je suis en train d'écrire un tuto qui explique comment installer RoR sur IIS.

mon testeur fétiche (merci a lui 😊) est derrière un proxy. Il ne peut donc pas installer RoR avec la commande

```
gem install rails --include-dependencies
```

et même en utilisant la commande

```
gem install rails -p http://<proxy-url>
```

il n'arrivait pas installer rails donc je lui ai donc conseillé de l'installer manuellement.



son retard (hormis pour l'ECMAScript) varie de 16 à 41% selon les standards, et Internet Explorer 7 est encore loin du compte malgré quelques améliorations...

On notera également la petite avance de Firefox pour les CSS 3 (qui ne sont pas finalisé pour la plupart).

Au final cela donne les résultats suivant :

	IE 6	IE 7	Firefox 1.5	Opera 8.5
HTML / XHTML	75%	75%	91%	85%
CSS 2.1	52%	59%	93%	93%
CSS 3 changes	10%	16%	27%	8%
DOM	50%	50%	79%	77%
ECMAScript	100%	100%	Y	Y

Vous pouvez retrouver les résultats détaillés sur la page suivante : [Web browser standards support summary](#)

Il est également possible d'[afficher d'autres navigateurs](#) (IE 4, IE 5, IE 5.5, Firefox 1.0, Konqueror 3.4 et Safari 2.0), mais les informations sont parfois incomplètes...

Retrouvez le blog d'adiGuba en ligne [[Lien55](#)]

Télécharger tous les fichiers .gem et non les autres fichiers comme .zip. Placer ces fichiers dans même répertoire. Par exemple gem. Il faut installer suivre cet ordre pour installer les gem.

1. [Activesupport](#)
2. [Actionpack](#)
3. [Actionmailer](#)
4. [Activerecord](#)
5. [Actionwebservice](#)
6. [Rake](#)
7. [Rails](#)

Dans une fenêtre Dos lancer cette commande

```
gem install gemname
```

Retrouvez le blog de bolo en ligne [[Lien56](#)]



Les derniers tutoriels et articles

Créer une application web PHP avec XMLRAD

1. Introduction

Pour créer une application web en PHP on utilise généralement un moteur de template comme Smarty ou PhpLib. Ces templates permettent la séparation du code PHP lié aux règles de gestion et la garniture HTML des pages facilitant le développement et la maintenance. Cependant, les moteurs de templates introduisent la plupart du temps une nouvelle syntaxe propriétaire aux moteurs.

[XMLRAD\[Lien57\]](#) est un framework qui se base sur la technologie standard XML/XSL pour générer les pages HTML. Elle offre la séparation des données (XML) de leur représentation (XSL). Le code de gestion est lui séparé dans un fichier .php. L'accès aux données est, par contre, pris en charge par le framework XMLRAD et les requêtes SQL sont décrites dans des fichiers XML. Le couple XML/XSL est ainsi une autre méthode de template s'appuyant sur un langage normalisé.

Nous allons voir dans cet article comment développer notre application web avec XMLRAD: Création du projet, connexion à la base de données, génération des premiers écrans et l'implémentation de règles de gestion. La création d'une application quelque soit votre environnement est décrite en détail dans l'article [XMLRAD: premiers pas\[Lien58\]](#). Je vous invite à le parcourir avant de commencer celui là.

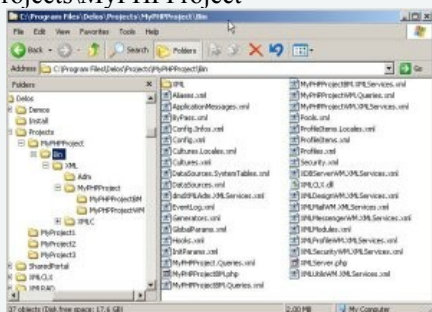
2. Création du projet

Lancez XMLRAD, et cliquez sur Create a new project. Sélectionnez Simple project.



Nommez votre projet MyPHPProject et sélectionnez le langage PHP, la version (4 ou 5) et le serveur web (IIS, apache 1.3 ou 2.0).

Créez votre projet avec des XMLModules normalisés. Votre projet est alors généré sur disque dans le répertoire C:\Program Files\Delos\Projects\MyPHPProject



Dans le répertoire du projet vous avez un répertoire Bin dans lequel

se trouve toute l'application. Un répertoire Data sera automatiquement créé pour toutes les traces, logs d'erreur, profiles, etc.

Dans ce répertoire Bin se trouve tous les fichiers de configuration XML:

- Config.xml : Fichier principal de l'application. Importe les fichiers InitParams.xml, GlobalParams.xml, XMLModules.xml, DataSources.xml, Aliases.xml et Pools.xml
- InitParams.xml : Fichier de paramétrages de l'application et du framework. voir [Appendix 14: InitParams and GlobalParams description \[Lien59\]](#)
- GlobalParams.xml : Fichier de constantes de l'application et du framework. voir [Appendix 14: InitParams and GlobalParams description\[Lien59\]](#)
- XMLModules.xml : Fichier de description des modules de l'application et du framework.
- DataSources.xml : Fichier de configuration de l'accès aux bases de données. voir [Appendix 3: Configuring DataSources\[Lien60\]](#)
- Aliases.xml : Fichier de configuration de l'alias de l'application utilisé dans les fichiers XSL.
- Pools.xml : Fichier de configuration des pools.
- ApplicationMessages.xml : Fichier contenant tous les messages d'applications traduits.
- ByPass.xml : Fichier de configuration des actions et des adresses IP qui ne sont pas soumises à la sécurité.
- Config.Infos.xml : Fichier stockant les informations du projet.
- Cultures.xml : Fichier de configuration des cultures (formats de dates, heures, nombres)
- dmdXMLAdm.XMLServices.xml : Fichier de description des XMLServices du module dmdXMLAdm. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)
- EventLog.xml : Fichier de configuration des traces.
- Generators.xml : Fichier de configuration des générateurs d'identifiants.
- Hooks.xml: Fichier de configuration des hooks.
- ProfileItems.xml : Fichier de description des éléments de profiles.
- Profiles.xml : Fichier des profiles par défaut.
- Security.xml : Fichier de configuration de la sécurité intégrée.
- XMLDesignWM.XMLServices.xml : Fichier de description des XMLServices du module XMLDesignWM. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)
- XMLMailWM.XMLServices.xml : Fichier de description des XMLServices du module XMLMailWM. voir [Appendix 9: XMLCLX reserved keywords and reserved](#)

[actions\[Lien61\]](#)

- XMLMessengerWM.XMLServices.xml : Fichier de description des XMLServices du module XMLMessengerWM. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)
- XMLProfileWM.XMLServices.xml : Fichier de description des XMLServices du module XMLProfileWM. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)
- XMLSecurityWM.XMLServices.xml : Fichier de description des XMLServices du module XMLSecurityWM. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)

- XMLUtilsWM.XMLServices.xml : Fichier de description des XMLServices du module XMLUtilsWM. voir [Appendix 9: XMLCLX reserved keywords and reserved actions\[Lien61\]](#)

On y trouve aussi les fichiers sources.

Dans le sous-répertoire bin se trouve une dll: XMLCLX.dll qui est la dll du framework XMLRAD chargée par l'extension php php_xmlrad.dll.

le sous-répertoire XML contient tous les fichiers XML décrivant les actions (XMLServices) et les fichiers XSL.

Retrouvez la suite de l'article de Jean philippe en ligne [[Lien62](#)]

Livres



XML en concentré

782 pages, 3ème édition, 21 avril 2005 Editions O'Reilly, ISBN: 2-84177-353-1
Public visé : Tous niveaux
Commandez sur [Amazon.de](#)



La quatrième partie, d'environ 300 pages, constitue le manuel de référence. Ce manuel couvre XML 1.0 et 1.1 ainsi que la plupart des technologies XML les plus abouties (et utilisées) à ce jour. Ce manuel est indispensable à tout programmeur souhaitant retrouver une syntaxe.

XML s'est imposé dans la plupart des champs de l'informatique moderne et est un outil de travail difficilement contournable pour les développeurs et les webmasters.

Cette troisième édition de XML in et Nutsbell, renommée XML en concentré, est une référence aussi concise que possible sur les principaux formats et API utilisés lors de l'analyse, la transformation et le rendu des sources de données XML.

Ceux qui sont concernés par l'échange de données seront conquis par les chapitres présentant la conception d'un document XML, les DTD, les espaces de noms, l'internationalisation, la présentation via DockBook, OpenOffice ou WordprocessingXML, et XML Schema et RelaxNG. Les webmasters tireront profit des chapitres décrivant l'utilisation de CSS, des jeux de caractères, XLink, XPointer, XInclude, ou encore RDDL, XPath, XSLT ou XSL-FO. Les développeurs auront sous la main non seulement la description complète de la norme XML 1.0 et 1.1, du format des DTD, des différents jeux de caractères, y compris Unicode, et des espaces de noms, mais également des API fondamentales comme SAX2 (Simple API, for XML) et DOM (Document Object Model) niveau 1 à 3. Les auteurs se sont attachés à ne parler que des langages, API et formats arrivés à maturité et pour lesquels il existe des outils ayant fait leur preuve. Pour chaque technique abordée, vous trouverez un exposé exhaustif de la syntaxe et des API, des exemples d'utilisation concrets. XML en concentré est bien plus qu'un simple aide-mémoire. Il est la référence indispensable pour quiconque travaille de près ou de loin avec ce format de document.

En bref, beaucoup de rigueur et un bon niveau technique pour cet ouvrage. Un livre bien pensé, pour les débutants qui s'accroche comme pour ceux ayant déjà des connaissances en la matière. On pourra regretter cependant l'absence de chapitres consacrés à SVG ou MathML.

(avis d'un débutant en XML)

Critique du livre par la rédaction ([Eric Grimois](#)) :

Ce livre est impressionnant, et est bien plus qu'un "simple" livre de référence car il offre, dans ses deux premières parties, une vue d'ensemble assez détaillée des technologies gravitant tout autour de XML. Il les illustre par de nombreux exemples, et il regorge également de conseils méthodologiques, particulièrement précieux pour ceux qui ne savent pas trop comment aborder la mise en oeuvre de ces technologies, et tout aussi profitables aux initiés pour améliorer leur pratique. Il est à noter que la partie référence ne reprend pas tous les sujets évoqués dans les deux premières - l'ouvrage doublerait de volume ! - mais elle contient toutes les informations primordiales pour le développeur souhaitant intégrer XML dans ses applications.

Comme dans la plupart des ouvrages édités par O'Reilly, les auteurs se focalisent sur les spécifications émises par le W3C. Bien que tout à fait légitime, il aurait été intéressant, vue l'ambition d'exhaustivité affichée par les auteurs, que cette primauté soit assouplie et que soient abordés d'autres technologies très présentes dans l'industrie et émanant d'autres instances, comme Relax NG (consortium OASIS) par exemple.

Critique du livre par la rédaction ([Nouridine Falola](#)) :

Cet ouvrage est composé de quatre parties. Les trois premières concernent l'apprentissage du XML et des technologies gravitant autour du XML. Assez bien adaptées aux débutants (d'aucun pourra trouver à juste titre ce livre un peu ardu pour commencer - dans ce cas on pourra trouver son bonheur avec "Introduction à XML"), ces parties ne seront pas inutiles en tant que révisions et approfondissement pour les connaisseurs. Les chapitres sont bien fournis en exemples souvent concis, et rigoureux tant la syntaxe doit être respectée (et donc comprise!).

Rédigé dans un langage clair et précis, cet ouvrage ne peut cependant constituer la base unique d'une autoformation à XML pour un débutant. Centré sur les technologies elles-mêmes, il n'évoque qu'assez brièvement la mise en oeuvre des outils (parseurs, processeurs, librairies, etc.) qui les implémentent. Je le conseille donc aux débutants comme le complément idéal d'un ouvrage d'initiation, et à tous les autres sans réserve.

Les derniers tutoriels et articles

Différentes façon de tracer l'activité de sessions

La surveillance de sessions est un élément fondamental dans la recherche de meilleurs performances. Vous allez voir ici quelques techniques possibles.

1. Introduction

Toutes les traces seront générées sur le serveur, dans le répertoire destiné aux fichiers générés par les processus "USER".

Ce répertoire est défini par le paramètre d'initialisation "user_dump_dest" que vous pouvez retrouver en faisant :

```
select value
  from v$parameter
 where Name = 'user_dump_dest';
```

Une fois la trace générée, il vous est possible de la lire directement (car c'est un fichier texte) mais l'utilitaire TKPROF vous permet de mettre en forme les résultats en effectuant ou non des opérations de tri/regroupement. [Jaouad\[Lien63\]](#) a écrit à ce sujet un [article\[Lien64\]](#) très complet !

Documentation Oracle :  [Formatting Trace Files with TKPROF\[Lien65\]](#)

2. Présentation des différentes techniques

Il existe plusieurs moyen de tracer les sessions Oracle, la différence se situant dans le niveau de détail et les éléments collectés.

2.1. SQL_TRACE

C'est le mode par défaut, les éléments collectés sont "basiques" : instructions SQL et quelques indicateurs chiffrés (temps CPU, temps écoulé, lectures physiques, ...)

Documentation Oracle :  [SQL_TRACE\[Lien66\]](#)

2.2. Event 10046 : SQL_Trace et plus si affinités

Cet évènement dérive du SQL_TRACE mais apporte des niveaux

de détails :

- **Niveau 1** : Strictement équivalent à SQL_TRACE
- **Niveau 4** : SQL_TRACE + valeur des bind variables
- **Niveau 8** : SQL_TRACE + informations sur les waits
- **Niveau 12** : SQL_TRACE + valeur des bind variables + informations sur les waits

"Documentation" Oracle :  [EVENT\[Lien67\]](#)

Sous Unix, le fichier

`$ORACLE_HOME/rdbms/mesg/oraus.msg` contient la liste de tous les events avec une description très sommaire. Ici, pour cet event, on a : 10046, 00000, "enable SQL statement timing"

En général, vous ne devez activer ces events que sur demande express du support !



2.3. Event 10053 : CBO computations

Cet event permet de surveiller plus particulièrement les plans d'exécutions utilisés par les différentes instructions SQL.

Il y a deux niveaux possible (1 et 2) et bizarrement, le niveau 2 est le moins complet. L'explain plan n'est affiché qu'en mode CBO; en mode RBO, seule la requête est tracée.

A propos de cet event, on a 10053, 00000, "CBO Enable optimizer trace".

Attention : le niveau d'informations présentés est très bas, les traces générés sont donc volumineuses. A titre d'exemple, un simple "SELECT * FROM DUAL" a généré une trace de 800 lignes (et de plus de 30 Ko). De plus, il semblerait que cet event ne soit déclenchable que pour la session courante....



Retrouvez la suite de l'article de Leo Anderson [\[Lien68\]](#)

Sécurisation d'une base Oracle

1. Préambule

Ce document présente différentes techniques pour sécuriser les données d'une base Oracle, c'est-à-dire empêcher leur consultation/modification par des personnes non explicitement autorisées à le faire. La sécurité, c'est assurer la disponibilité, l'intégrité et la non divulgation des données. Oracle étant une application tournant sur un système, la sécurisation Oracle n'a de sens que sur un système sécurisé. La sécurisation d'un OS étant très vaste et très variable selon les OS et les contextes, nous nous contenterons de prendre comme hypothèse que le système qui héberge les bases Oracle est sécurisé.

Vous l'aurez compris, ce document nécessite d'avoir de bonnes connaissances Oracle ainsi qu'une certaine culture informatique pour pouvoir être abordé efficacement. Les notions présentées le sont ici dans leur grandes lignes pour vous permettre d'en comprendre la portée et le rôle global; si vous souhaitez approfondir quelques points précis, la documentation officielle Oracle devrait alors vous permettre de trouver ce que vous cherchez.



Sauf mention contraire, aucune des notions présentées dans ce document n'est relative à une version précise (tout du moins parmi les versions récentes 8i, 9i et 10g).

Retrouvez la suite de l'article de Leo Anderson en ligne [\[Lien69\]](#)

Les derniers tutoriels et articles

Gestion de la roulette de la souris dans les formulaires

Gestion de la roulette de la souris dans les formulaires.
Ajoutez un événement MouseWheel à vos formulaire et personnalisez l'action de la roulette.

1. Introduction

Utiliser la roulette de la souris dans un formulaire entraîne un changement d'enregistrement qui peut être indésirable.

En effet un utilisateur peut s'attendre à faire défiler le formulaire vers le bas, mais l'action sur la roulette de la souris passe à l'enregistrement suivant.

Cela peut être agaçant, voir très problématique car l'enregistrement est validé alors que l'utilisateur ne le souhaitait pas.

Pour pallier ce comportement on peut utiliser une librairie activeX qui va nous fournir un nouvel événement lors de l'action de la roulette

2. La librairie dll fournie par Microsoft

Microsoft fourni [ICI\[Lien70\]](#) une dll qui permet d'annuler l'action de la roulette de la souris. Malheureusement le code de cette dll n'est pas très fiable et ne fonctionne pas sur Access 2003.

Voici la liste des disfonctionnements que l'on a pû relever :

- incompatibilité avec Access 2003 (apparemment conflit de nom avec le nouvel événement Mousewheel);
- l'utilisation de la dll dans un sous-formulaire et son parent rend le sous-formulaire inaccessible;
- la fermeture du formulaire lors d'un aperçu avant impression fait crasher l'application;
- l'utilisation de la dll dans plusieurs formulaires est mal gérée, l'événement est renvoyé dans le dernier formulaire uniquement.
--> il n'est donc pas possible de conditionner l'action de la roulette dans les autres formulaires car ils ne reçoivent pas l'événement;

Et il manque à mon goût une information utile dans l'événement :

- dans quel sens a été déplacée la roulette? Cette information est utile si on désire exécuter une action spécifique.

C'est pour toutes ces raisons que j'ai développé une nouvelle librairie dll, baptisée pour l'occasion MouseWheelDVP.

3. Référencer la librairie MouseWheelDVP?

L'utilisation reste pratiquement identique à celle de la dll originale de Microsoft.

C'est une dll activeX, ce qui signifie qu'elle doit être enregistrée et que la référence doit être sélectionnée dans Access.



Attention : une fois enregistrée la dll ne doit pas changer d'emplacement.

Si vous désirez changer la dll d'emplacement, désenregistrez la, puis référez la à nouveau depuis le nouvel emplacement.

3.1. Enregistrement dans Access

Pour référencer la dll :

- allez dans l'éditeur VBA : Outils --> Macro --> Visual Basic Editor ; ou ALT+F11; ou Affichage --> Code sous Access 97
- allez dans le menu : Outils --> Références;
- cliquez sur le bouton parcourir et sélectionnez le fichier MouseWheelDVP.dll;
- > la librairie est automatiquement enregistrée dans le registre de Windows en suivant cette procédure.

3.2. Enregistrement manuel

- cliquez sur le menu Démarrer de Windows, puis Exécuter.
- tapez la commande suivante :
 - pour enregistrer :
regsvr32.exe C:\VotreChemin\MouseWheelDVP.dll
 - pour désenregistrer :
regsvr32.exe /u C:\VotreChemin\MouseWheelDVP.dll

Remarque : Il faut également cocher ou décocher la référence dans Access.

3.3. Enregistrement par le code VBA

Il est possible d'enregistrer la librairie avec des fonctions VBA.

Exemple de code à placer dans un module :

Fonctions d'enregistrement de librairies

```
Option Compare Database
Option Explicit

Private Declare Function LoadLibrary Lib "kernel32" Alias "LoadLibraryA"
(ByVal lpLibFileName As String) As Long
Private Declare Function FreeLibrary Lib "kernel32" (ByVal hLibModule As
Long) As Long
Private Declare Function GetLongPathName Lib "kernel32" Alias
"GetLongPathNameA" _
(ByVal lpszShortPath As String, ByVal lpszLongPath
As String, _
ByVal cchBuffer As Long) As Long

Private Declare Function DVPDllRegisterServer Lib "MouseWheelDVP" Alias
"DllRegisterServer" () As Long
Private Declare Function DVPDllUnregisterServer Lib "MouseWheelDVP" Alias
"DllUnregisterServer" () As Long
Private Declare Function DVPDllCanUnloadNow Lib "MouseWheelDVP" Alias
"DllCanUnloadNow" () As Long

' Enregistre la librairie
Private Sub FnRegLib()
Dim lLib As Long ' Identifiant de la librairie
Dim lReg As Long ' Pointeur de la fonction d'enregistrement
' Référence la librairie dans Access
On Error Resume Next ' si la librairie est déjà référencée on essaye
quand même de l'enregistrer
Application.References.AddFromFile ApplicationPath &
"MouseWheelDVP.dll"
On Error GoTo Gestion_Erreurs
' Charge la librairie
lLib = LoadLibrary(ApplicationPath & "MouseWheelDVP.dll")
If lLib = 0 Then
MsgBox "Impossible de trouver la librairie : " & vbCrLf &
```

```

ApplicationPath & "MouseWheelDVP.dll"
Exit Sub
End If
' Enregistre la librairie (en plus du AddFromFile qui parfois
n'enregistre pas correctement la librairie)
If DVPDllRegisterServer <> 0 Then
MsgBox "Erreur lors du référencement de la librairie"
End If
Gestion_Erreurs:
If Err.Number <> 0 Then MsgBox Err.Description
' Libère la librairie
FreeLibrary lLib
End Sub

' Désenregistre la librairie
Private Sub FnUnregLib()
Dim lLib As Long ' Identifiant de la librairie
Dim lCanUnload As Long ' Pointeur de la fonction de test d'utilisation
de la dll
Dim lUnReg As Long ' Pointeur de la fonction d'enregistrement
' Supprime la référence dans Access
On Error Resume Next ' si la librairie n'est pas référencée on essaye
quand même de la désenregistrer
Application.References.Remove
Application.References.Item("MouseWheelDVP")
On Error GoTo Gestion_Erreurs
' Charge la librairie
lLib = LoadLibrary(ApplicationPath & "MouseWheelDVP.dll")
If lLib = 0 Then
MsgBox "Impossible de trouver la librairie :" & vbCrLf &
ApplicationPath & "MouseWheelDVP.dll"
Exit Sub
End If
' Vérifie que la librairie n'est pas en cours d'utilisation
If DVPDllCanUnloadNow <> 0 Then
MsgBox "Impossible de déréférencer la librairie maintenant" &
vbCrLf & "Quittez les formulaires utilisant la librairie"
GoTo Gestion_Erreurs
End If
' Désenregistre la librairie (elle disparaît alors de la liste des
références)
If DVPDllUnregisterServer <> 0 Then
MsgBox "Erreur lors du référencement de la librairie"
End If
Gestion_Erreurs:
If Err.Number <> 0 Then MsgBox Err.Description
' Libère la librairie
FreeLibrary lLib
End Sub

' Récupère le chemin de l'application (chemin long)

```

```

' (pour remplacer CurrentProject.Path dans Access 97)
Public Function ApplicationPath() As String
Dim lRet As Long
Dim lShortPathName As String
Dim lLongPathName As String
lLongPathName = Space(1024)
lShortPathName = Left(CurrentDb.Name, Len(CurrentDb.Name) -
Len(Dir(CurrentDb.Name)))
lRet = GetLongPathName(lShortPathName, lLongPathName,
Len(lLongPathName))
ApplicationPath = Left(lLongPathName, lRet)
End Function

```

Pour enregistrer la librairie qui se trouve dans le même répertoire que l'application :

Enregistrer la librairie

Call FnRegLib

Pour désenregistrer la librairie qui se trouve dans le même répertoire que l'application :

Désenregistrer la librairie

Call FnUnregLib

3.4. Enregistrement de la librairie sous Windows 98

Si vous êtes sous Windows 98 vous allez peut-être rencontrer quelques difficultés pour enregistrer la dll.

Celle-ci étant développée en VB6, installez le runtime VB6 pour pouvoir l'enregistrer.

Retrouvez la suite de l'article de Thierry Gasperment en ligne [[Lien71](#)]

Formulaire de recherche prêt à l'emploi 2ème partie.

Cet article est le deuxième de la série consacrée à la recherche dans Microsoft ACCESS. Au terme du [précédent article](#) nous avons créé un formulaire de recherche générique. Dans celui-ci nous allons agrémenter notre formulaire de plusieurs fonctions importantes.

1. Avertissement



L'utilisation de la touche F1 est vivement conseillée à tous les stades de l'utilisation d'ACCESS. L'amélioration constante de l'aide en fait un partenaire de choix dans l'apprentissage permanent d'ACCESS. Personnellement, je ne peux m'en passer, ne serait-ce que pour mémoire.

2. Conseils importants sur la programmation

Suite à des remarques sur le tutoriel précédent, je tiens à donner quelques conseils sur la manière de nommer certains objets dans Microsoft ACCESS comme les formulaires, états, tables et champs.

Même si l'utilisation d'espaces dans les noms de tables et de champs est autorisée dans Microsoft ACCESS, il est fortement **déconseillé** de le faire. En effet chaque fois qu'un nom comporte au moins un espace il faut l'entourer de crochets []. Dans le QBE (générateur de requête) ces crochets peuvent être insérés automatiquement. Mais le QBE peut interpréter également le nom comme une chaîne de caractères ou au pire provoquer une erreur. Dans le code VBA il faut toujours le faire manuellement ce qui ralentit l'écriture du code et peut causer des erreurs.

Remplacez l'espace par l'underscore _.



Lisez l'excellent tutoriel d'[Argyronet](#) sur les conventions d'écriture dans VBA.

Pour éviter des problèmes avec les espaces et autres caractères veuillez appliquer cette modification dans **Cmd_recherche_click()**

Code à modifier

```

strTable = Me.cbo_Table ' recupère le nom de la table
strField = Me.cbo_Champ ' recupère le nom du champ

```

Code modifié

```

strTable = "[" & Me.cbo_Table & "]" ' recupère le nom
de la table
strField = "[" & Me.cbo_Champ & "]" ' recupère le nom
du champ

```

3. Présentation des nouvelles fonctionnalités

Les fonctionnalités ajoutées sont les suivantes :

1. Visualisation du code SQL généré
2. Gestion du code SQL
3. Traitement des opérateurs logiques principaux (AND, OR)
4. Traitement de la valeur Null
5. Ouverture d'un formulaire de visualisation
6. Choix des champs à visualiser dans la liste de résultat

Retrouvez la suite de l'article de Fabrice Constans en ligne [[Lien72](#)]

Les derniers tutoriels et articles

Publication d'applications AJAX

1. Introduction

La presse relaye régulièrement l'intérêt des développeurs pour **AJAX**, fer de lance du Web 2.0.

Le « client riche » se cherche mais la simplicité du modèle séduit : Capitaliser sur les avantages du Web (standardisation forte, légèreté du déploiement, ...) en améliorant l'interface utilisateur des applications. Bien qu'encore peu présente en production, les initiatives autour d'**AJAX** gagnent chaque jour en maturité et les grands éditeurs ont rejoint le rang des premiers fidèles du monde libre. Certains Framework émergent et il n'est pas rare aujourd'hui de « saupoudrer » un peu d'**AJAX** dans les sites Web existants.

Ce document se propose de donner les éléments techniques permettant de mettre en oeuvre AJAX dans les sites Web publiés par 4ème Dimension. Il complète ainsi les bases de démonstration et notes techniques disponibles sur le sujet.

2. Un échange Client-Serveur

AJAX (Asynchronous Javascript And XML) est une technique de développement.

Ce n'est pas une technologie en soit mais un regroupement sous un même nom de plusieurs standards du Web :

- **XHTML** et **CSS** pour la création de l'interface utilisateur
- **DOM** pour la manipulation dynamique de cette interface
- **XML** et **XSLT** pour l'échange d'information avec le serveur
- **JavaScript** pour la logique applicative

L'élément permettant au navigateur de communiquer avec le serveur 4ème Dimension est présent dans Javascript sous forme d'un objet nommé **XMLHttpRequest**. Sa mise en oeuvre est détaillée page 5 et 6.

Les informations circulant entre le client et le serveur seront généralement proposés sous forme XML. Les possibilités de 4ème Dimension en la matière sont détaillés chapitre V.

4. L'objet XMLHttpRequest

XMLHttpRequest est un objet JavaScript permettant d'exécuter des requêtes HTTP. Après traitement par le serveur, cet objet reçoit une réponse HTTP sous forme d'une chaîne de caractères ou d'un objet XML manipulable par le DOM du navigateur.

XMLHttpRequest n'étant pas normalisé on trouve différentes implémentations au sein des navigateurs nécessitant un ensemble de tests préalables à son utilisation.

Prototype de fonction retournant l'objet **XMLHttpRequest** adéquat en fonction du navigateur :

```

{
var req = false;
try
{
req=new XMLHttpRequest();
}
catch(e)
{
try
{
req=new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e)
{
try
{
req = new ActiveXObject("Microsoft.XMLHTTP");
}
catch(e)
{
req = false;
}
}
}
return req;
}

```

Retrouvez la suite de l'article de Gérald Czwiklinski en ligne [\[Lien73\]](#)

Tips

Compter le nombre d'occurrences dans un texte

Une méthode pour compter le nombre d'occurrences d'une chaîne dans un texte :

Exemple avec :

str_gLon_Nbr_Occurences("un champ texte, un champ alpha et un champ entier long";"un")

```

-----
` Méthode : str_gLon_Nbr_Occurences
` Créée le 22/12/05 par Vincent de Lachaux
-----

```

```

` Description
` Retourne le nombre d'occurrences de $2 dans $1
-----

```

C_TEXTE(\$1) `Chaîne dans laquelle chercher
C_TEXTE(\$2) `Chaîne à trouver
C_ENTIER LONG(\$0) `Nombre d'occurrences

\$0:=(Longueur(\$1)-Longueur(Remplacer chaîne(\$1;\$2;"")))/Longueur(\$2)

Les pièges du C

1. Introduction

Le C est un langage proche de la machine (manipulation d'adresses mémoire) et le compilateur laisse une grande liberté au développeur. De ce fait, écrire un programme en C demande beaucoup de rigueur. Cependant écrire un code propre et sans comportement indéfini n'est pas suffisant. En effet, l'intérêt pour l'informatique serait bien moindre sans l'interactivité homme-machine. Hélas s'il y a un point sur lequel on ne peut faire aucune supposition c'est bien la réponse que va fournir un utilisateur à notre programme. Dans un monde idéal où tous les utilisateurs seraient voués de bonnes intentions, une mauvaise réponse entraînerait dans le pire des cas, un arrêt brutal du programme et une perte des données non enregistrées. Mais comme nous ne sommes pas dans un monde parfait, il existe des utilisateurs mal intentionnés et les conséquences peuvent être bien plus graves qu'un simple plantage. Et bien souvent le but recherché par les pirates est d'obtenir les pleins droits sur une machine (un shell avec les droits root sur un système de type Linux par exemple) ou encore lancer l'exécution d'un virus. Maintenant que j'ai essayé de vous convaincre de la dangerosité d'un programme possédant des failles, je vais vous décrire les types de failles les plus connus et surtout la manière de les éviter. Il est peut être un peu tard mais je m'excuse auprès des personnes qui lisent cet article dans le but de pirater un ordinateur, en effet il n'y a aucune explication à ce sujet et histoire de vous encourager à aller voir de quoi il s'agit, sachez que pour espérer mettre ceci en pratique, il faut avoir quelques connaissances en [C](#) (audit de code), en [assembleurs\[Lien74\]](#) et en [conceptions des systèmes d'exploitation\[Lien75\]](#) : bon courage ;)

2. Dépassement de tampon

Aussi appelé buffer overflow, il s'agit sûrement du type de faille le plus connu et le plus simple à éviter. Lors d'un dépassement de tampon, l'utilisateur entre une chaîne de caractères qui va se retrouver dans un tableau de caractères de taille insuffisante ce qui va entraîner l'écriture de données en dehors de la zone mémoire allouée pour le tableau. Suivant le type de tableau (statique ou dynamique), on distingue deux types de buffer overflow :

- Stack overflow : dépassement de tampon qui va écraser la pile (tableau en mémoire statique)
- Heap overflow : dépassement de tampon qui va écraser le tas (tableau dynamique)

2.1. Sur la pile

Pour commencer voici un petit code qui illustre le problème :

```
#include <stdio.h>
#include <string.h>
static void cpy_arg (const char *s)
int main (int argc, char **argv)
{
    if (argc == 2)
    {
        cpy_arg (argv[1]);
        printf ("argv[1] copier\n");
    }
}
```

```
return 0;
}
static void cpy_arg (const char *s)
{
    if (s)
    {
        char text[20];
        strcpy (text, s);
    }
}
```

Les habitués auront tout de suite repéré que si l'utilisateur fournit comme argument à notre programme une chaîne de plus de 19 caractères, on obtient un dépassement de capacité lors de la copie dans *text*. Si le dépassement est suffisamment important, il est possible d'écraser des données système stockées dans la pile. Cette zone de mémoire doit son nom à la manière dont elle est utilisée : les données sont empilées lors de l'appel d'une fonction puis dépilées dans l'ordre inverse au retour dans la fonction appelante. Les données stockées dans la pile sont :

- Les arguments de la fonction appelée
- Ses variables locales
- La valeurs des registres processeurs qui vont permettre, lors du retour de la fonction, de retrouver le contexte d'exécution de la fonction appelante

Ce sont ces registres qui vont nous intéresser puisque l'un d'entre eux est destiné à stocker l'adresse de la prochaine instruction à exécuter. Maintenant imaginez que dans notre fonction *cpy_arg*, on modifie cette valeur, au retour dans le *main*, le programme, à la place d'exécuter l'instruction *printf*, va chercher une instruction à l'adresse modifiée, qui peut pointer sur une zone mémoire où un pirate aura préalablement stocké un morceau de code de son cru (appelé *shellcode*). Ceci peut être facilement démontré :

```
$ gcc -Wall -W -O2 -ansi -pedantic bof.c
$ gdb a.out
...
(gdb) set args `perl -e "{print 'A'x512}"`
(gdb) run
Starting program: ./a.out `perl -e "{print 'A'x512}"`
Program received signal SIGSEGV, Segmentation fault.
0x41414141 in ?? ()
(gdb)
```

Notre programme plante car il a essayé d'exécuter une instruction à l'adresse 0x414141 (0x41 est le code ASCII du caractère 'A'). On pourrait minimiser les risques d'une telle faille en se disant qu'il faut pouvoir entrer assez de données pour atteindre la pile. Hélas un débordement d'un byte peut permettre d'arriver au même résultat, on parle alors de *off-by-one overflow* (dans ce cas, plutôt que de modifier la valeur d'un registre placé dans la pile, on modifie la valeur d'une variable locale).

Pour conclure cette première partie, je vous propose un code que je trouve assez amusant et qui montre les véritables dangers d'une telle faille :

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

```

static void cpy_arg (const char *);
void code_pirate (void);
int main (int argc, char **argv)
{
    if (argc == 2)
    {
        printf ("code_pirate = %p\n", code_pirate);
        cpy_arg (argv[1]);
        printf ("argv[1] copier\n");
    }
    return 0;
}
static void cpy_arg (const char *s)
{
    if (s)
    {
        char text[20];
        strcpy (text, s);
    }
}
void code_pirate (void)
{
    printf ("Code pirate execute !\n");
}

```

On l'exécute :

```

$ a.out aaaabbbbccccddddeeeeffffggghhhhhiiiijjjjkkkk\x23\x13\x40
code_pirate = 00401323
Code pirate execute !
Segmentation fault

```

Et oh stupeur, la fonction `code_pirate` qui n'est jamais appelée dans notre code se retrouve exécutée ! Tout simplement parce qu'à la place d'écraser la valeur de la prochaine instruction à exécuter par une valeur quelconque (0x414141), on la remplace par l'adresse de la fonction `code_pirate` (little endian oblige, les octets sont entrés à l'envers).

Maintenant imaginez que l'adresse entrée n'est pas celle d'une fonction de notre programme mais celle d'une fonction présente dans une bibliothèque à liaison dynamique !

Dans ce cas d'école, le programme plante mais il est possible de le faire se terminer normalement : l'exécution du code est alors transparente pour l'utilisateur.

2.2. Sur le tas

Dans notre cas précédent, nous avons travaillé sur un tableau alloué en mémoire statique. Comme il existe des tableaux alloués en mémoire dynamique (fonction `malloc`, `calloc` et `realloc`) il existe aussi un dépassement de tampon qui va avec : le *heap overflow*. Cette fois le principe est un peu plus compliqué puisqu'il s'agit de modifier le contenu des structures de données utilisées par le système lors d'une allocation/désallocation de mémoire pour y stocker le code malicieux.

3. Format de chaîne de caractères

Qu'est ce qu'un format de chaîne de caractères ? Il s'agit du premier argument de la fonction `printf`, qui permet de spécifier le format de la chaîne de caractère à afficher (ce paramètre est présent chez toutes les fonctions de la famille de `printf`). Comme à chaque fois, le problème survient lorsque l'utilisateur a le contrôle sur le contenu de cette chaîne :

```

#include <stdio.h>
int main (int argc, char **argv)
{
    if (argc == 2)
    {
        char text[256];
        sprintf (text, "%s\n", argv[1]);
        printf (text);
    }
}

```

```

}
return 0;
}

```

Le spécificateur de format `%x` permet d'afficher une valeur en hexadécimale :

```

$ ./a.out "%x"
8048524

```

En l'absence de paramètre, la fonction `printf` va afficher les valeurs présentes dans sa pile d'appel. Voici un exemple plus parlant :

```

$ ./a.out "AAAA|%x|%x|%x|%x|"
AAAAA|8048524|bffc1ac3|1|41414141|

```

Au bout d'un moment, on retrouve le début de notre chaîne ! Tout cela n'aurait pas de conséquences désagréables s'il n'était pas possible d'écrire en mémoire. Hélas la fonction `printf` propose l'option `%n` qui permet de sauvegarder le nombre de caractère déjà écrit :

```

#include <stdio.h>
int main (void)
{
    int i = 0;
    printf ("Hello World!\n\n", &i);
    printf ("Cette phrase contient %d caracteres\n", i);
    return 0;
}

```

Ce qui donne :

```

$ ./a.out
Hello World!
Cette phrase contient 13 caracteres

```

On a bien écrit la valeur 13 à l'adresse `&i`. Eh bien essayons :

```

$ gdb a.out
...
(gdb) set args "AAAA|%x|%x|%x|%x|\n"
(gdb) run
Starting program: ./a.out "AAAA|%x|%x|%x|%x|\n"
Program received signal SIGSEGV, Segmentation fault.
0xb7e4b8a3 in vfprintf () from /lib/tls/libc.so.6

```

4. Dépassement de capacité d'entier

Dernière vulnérabilité et sûrement la plus difficile à détecter. En effet cette faille utilise le fait que les nombres ne peuvent dépasser une certaine valeur qui dépend du nombre de bits utilisés pour leur représentation et de leur méthode de codage. Par exemple pour un **unsigned int** codé sur 32 bits en représentation binaire pur, la variable peut prendre une valeur allant de 0 à 4 294 967 295. La bibliothèque standard du C permet de retrouver ces valeurs grâce aux fichiers d'en-tête `limits.h` :

```

#include <stdio.h>
#include <limits.h>
int main (void)
{
    unsigned int ul = UINT_MAX;
    printf ("ul=%u\n", ul);
    return 0;
}

```

Ce qui donne sur ma machine :

```

ul=4294967295

```

Mais que se passe-t-il si l'on dépasse cette valeur ? Voici un exemple :

```

#include <stdio.h>

```

```
int main (void)
{
    unsigned int i = 1;
    while (i > 0)
    {
        i++;
    }
    printf("Hello world!\n");
    return 0;
}
```

D'un point de vu algorithmique, notre programme reste indéfiniment dans la boucle **while** mais si vous exécutez ce code suffisamment longtemps vous devriez voir s'afficher 'Hello world!'. Cela signifie que la valeur de *i* est devenue négative! Pour vous montrer la difficulté de détecter ce genre de faille, voici un code vulnérable :

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main (int argc, char **argv)
{
    int ret = EXIT_FAILURE;
    if (argc == 2)
    {
        char buf[255];
        unsigned char size = strlen (argv[1]);
        if (size < 254)
        {
            printf ("size = %d\n", size);
            strcpy (buf, argv[1]);
            ret = EXIT_SUCCESS;
        }
    }
    return ret;
}
```

Après l'avoir compilé avec le mode parano de gcc :

```
$ ./a.out `perl -e "{print 'A'x500}"`
size = 244
Erreur de segmentation
```

Eh oui, la valeur 500 stockée dans un **unsigned char** (codé sur 2 octets sur ma machine), se transforme en 244 et nous permet de copier 500 caractères dans une tableau prévu pour en contenir que 255 et l'on se retrouve dans le cas classique d'un [buffer overflow!](#)

5. Conclusion

Comme vous avez pu le constater, bien configurer son compilateur ne suffit pas pour obtenir du code sûr. Il faut aussi privilégier les fonctions qui tiennent compte de la taille de leurs arguments, telles que [fgets\[Lien76\]](#), [strncpy\[Lien77\]](#) ou encore [strncat\[Lien78\]](#) pour éviter tout débordement. Cependant pour ces deux dernières, il ne s'agit pas d'une solution idéale puisque que si la taille de la chaîne dépasse celle spécifiée en paramètre, le 0 de fin de chaîne n'est pas ajouté (il suffit de le faire manuellement, encore faut-il y penser). Il semble tout de même utopique de n'exécuter que des programmes dont le code est sans faille. Par conséquent, plutôt que passer en revue tous les codes (à supposer que le code soit disponible) pour le sécuriser afin d'éviter l'exécution de code malveillant, il faut penser à une autre solution : sécuriser l'environnement d'exécution des programmes.

Par exemple sous Linux, il existe un ensemble de patch pour le noyau : [GrSecurity\[Lien79\]](#) qui contient, entre autres, le patch PaX (Protection against eXecution) qui limite les conséquences des failles vu dans cette article. Bien sûr ce genre de protection n'est pas une excuse pour laisser des failles dans vos programmes puisqu'elle n'est pas invulnérable mais permet de rendre la tâche du pirate (beaucoup) plus compliquée.

Comme je l'ai précisé en introduction, je n'ai abordé que les failles les plus connues (de moi surtout) donc si vous en connaissez d'autres, je les intégrerai à cet article avec plaisir ;) Pour me contacter, vous trouverez un lien pour m'envoyer un MP sur la page de [l'équipe C & C++\[Lien80\]](#).

Retrouvez l'article de Nicolas Joseph en ligne [\[Lien81\]](#)

Livres



Effective C++ (3ème édition)

55 Specific Ways To Improve Your Programs And Designs

320 pages, 3 ème édition, mai 2005 Editions

Addison-Wesley Professional, ISBN: 0321334876

Commandez sur Amazon.fr

amazon.fr

Critique du livre par la rédaction ([Miles](#)) :

Cette critique s'applique à la troisième édition de ce livre (Le C++ efficace), en anglais. Le sous titre est [55 Specific Ways to Improve Your Programs and Designs](#). 5 éléments ont été apparemment ajoutés.

En fait, le livre a été complètement remanié, par exemple la première partie n'est plus consacrée à la migration C vers C++, mais à l'adaptation au C++ à travers les mots-clés const, les initialisations, ... La difficulté est croissante, les premiers éléments étant très simples à appréhender et à mettre en oeuvre - logique, c'est l'adaptation au C++ - puis on avance vers des terrains de plus en plus minés - qu'est-ce que le compilateur crée comme fonction

qu'on ne veut pas forcément, les opérateurs d'assignations, ... -. La partie sur la gestion de la mémoire gagne en complexité et s'applique à toutes les ressources, non plus seulement la mémoire. Les parties dédiées au design et déclaration des classes, leurs implémentations ainsi que l'héritage sont elles aussi pertinentes. Certains éléments des anciennes éditions ont été déplacés dans d'autres sections, d'autres éléments ont été incorporés judicieusement. Une partie dédiée à la programmation générique a été introduite, on va même jusqu'à la méta-programmation. La section consacrée à new et delete est aussi nouvelle et on parle de leur surcharge, ce qui est très intéressant.

En fait, ce livre est vraiment dédié au C++ et non à la jonction C/C++ comme par le passé. Il est même orienté vers le futur, plusieurs allusions au nouveau standard en discussion sont faites dans tout le livre avec les nouvelles classes apparues dans TR1:: et Boost. Ce livre, dans cette dernière édition, est de nouveau à la pointe, comme il l'a été lors de la première édition, après avoir marqué le pas face aux livres plus récents et plus "C++". Encore un livre à avoir dans sa bibliothèque !



Les derniers tutoriels et articles

Le Treeview et vous

Ce pratique expose et explique les principales propriétés et méthodes du Treeview.

La première section traite d'un Treeview non dépendant des données. Après une courte approche théorique visant à introduire le vocabulaire, il y est expliqué comment ajouter des noeuds de même niveau, parents ou enfants, et comment les gérer.

La seconde section expose comment construire, utiliser et sauvegarder un Treeview dépendant des données. Les données sont issues d'une base Access jointe au projet. La construction de ce Treeview utilise une méthode récursive afin de parcourir tous les enregistrements de la table et de les positionner en fonction de leur position hiérarchique et de dessiner les noeuds.

La troisième section montre comment déplacer un noeud du treeview et comment modifier l'enregistrement correspondant dans la table Access.

Un projet Visual Basic sous-tend le pratique.

Ce pratique, sans être très difficile, demande quand même une connaissance minimum de la programmation en VB et de l'accès aux données avec ADO. Le lecteur est guidé pas à pas du début à la fin du code, pratiquement chaque ligne étant commentée. Il lui est cependant recommandé de mettre en oeuvre de lui-même ce qui est exposé si il veut s'approprier réellement les connaissances utilisées.

Stocker et Annuler les mises à jour dans un formulaire

Thèmes abordés :

- . identification des événements liés aux mises à jour
- . codage d'une solution de stockage
- . codage d'une solution d'annulation
- . implémentation dans un formulaire

Niveau requis : **moyen / avancé**

1. Introduction



Le but de ce tutoriel est de proposer une solution de stockage et d'annulation de mise à jour de données dans Access.

Sur le forum, se trouvent de nombreux sujets abordant les problématiques suivantes :

- . comment savoir qui a modifié ma table ?
 - . comment connaître la dernière mise à jour de données ?
 - . comment annuler les mises à jour d'un utilisateur ?
 - . peut-on déclencher un code lors de la modification de données ?
- Ces questions récurrentes méritent qu'on s'y penche, et qu'on tente de leur apporter une réponse.

Ce champ de questions est issu d'une même problématique : **il n'y a pas dans Access et son moteur JET de triggers.**

Qu'est-ce qu'un **trigger** ?

Déclencheur (trigger)

Un déclencheur est une procédure stockée qui s'exécute lors d'une tentative d'action particulière (insertion, modification, suppression) sur une table ou sur une vue.

Plus d'informations sur ce site [Les Triggers en SQL\[Lien82\]](#).

Dans MS Access, les modifications / insertions / suppressions dans une table / requête n'occasionnent aucun événement et ne peuvent actionner de déclencheur (trigger), notre dernier recours est d'utiliser les **formulaires** qui sont aptes à manipuler ce type d'événements.

2. Méthodologie - Evénements

L'objectif est de pouvoir stocker les mises à jour et de permettre leur annulation.

Le tutoriel est prévu pour fonctionner sur un **formulaire lié à une source de données (dépendant)**. Dans le cas de formulaire indépendant, il suffirait de coder un traçage sur la sub de mise à jour.



- Pour annuler une mise à jour il nous faut :
- . cas d'une suppression : stocker les valeurs de l'ancien enregistrement ;
 - . cas d'une modification : stocker les anciennes valeurs qui ont été écrasées ;
 - . cas d'une insertion : stocker la clé du nouvel enregistrement en vue de son éventuelle suppression ;
 - . tous cas : enregistrer le login de la personne qui modifie, et l'heure de mise à jour.

Pour parvenir au but recherché, c'est à dire tracer toutes les mises à jour de données et permettre de les annuler, nous allons travailler sur les formulaires.

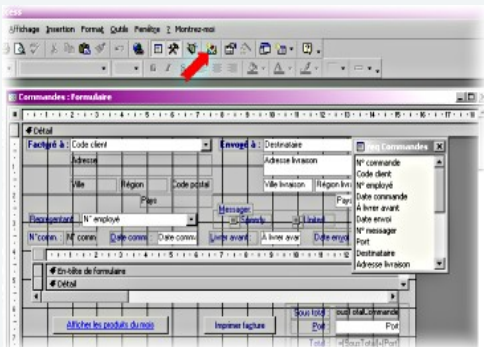
Ce tutoriel ne pourra pas fonctionner dans les cas suivants :

- . saisies directes dans des tables ou requêtes ;
- . imports de données externes ;
- . requête ajout ou mise à jour.

Pour stocker les mises à jour, nous devons déterminer quels événements activent ces modifications.

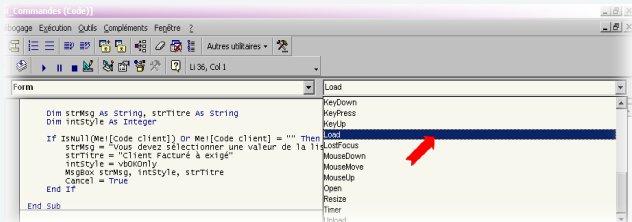
Nous travaillons sur la base **comptoir.mdb** fournie dans le répertoire *Samples* de Microsoft Office.

Nous ouvrons un formulaire en mode création, et nous explorons le code VBA.



Dans les deux listes déroulantes nous sélectionnons **Form** (désignant le formulaire) dans celle de gauche qui représente la liste des objets du formulaire.

Et nous parcourons celle de droite qui désigne les méthodes et événements.



Nous cherchons quels sont les événements les plus appropriés pour tracer les modifications.

. **traçage d'une insertion** : nous choisissons **AfterInsert**.

Seuls deux événements concernent un **INSERT**, Before et After. Si nous choisissons Before (avant) nous n'aurons pas accès aux nouvelles valeurs insérées, donc nous ne pourrions rien stocker.

Notre choix se porte sur **AfterInsert**, événement avec lequel nous connaîtrons les valeurs du nouvel enregistrement.

. **traçage d'une modification** : nous choisissons **BeforeUpdate**.

Ici aussi, deux événements prennent en charge un **UPDATE**, Before et After.

Si nous sélectionnons After, le jeu d'enregistrement aura été changé, et nous n'aurons plus accès aux anciennes valeurs, ce qui rend impossible toute annulation.

Notre choix se porte sur **BeforeUpdate** qui nous permet de travailler avec les anciennes valeurs (propriété **.OldValue**).

. **traçage d'une suppression** : nous n'avons pas d'autre événement que **Delete**.

3.Méthodologie - Stockage



Nous avons sélectionné les événements pertinents liés à une mise à jour quelle qu'elle soit dans la partie précédente.

Il nous faut maintenant choisir une méthode de stockage qui nous permette l'éventuelle annulation.

La méthode retenue peut faire débat, et je n'ai pas la certitude qu'elle soit nécessairement la meilleure.

Regardons l'éventail des possibilités :

. **duplication des données**

Si nous copions les anciennes données il sera facile de revenir en arrière. L'inconvénient est que cette méthode peut se révéler lourde, et qu'il faudrait dupliquer toutes les tables à surveiller.

. **stockage champ par champ**

Nous créons un enregistrement par champ modifié. L'inconvénient est qu'on ne peut annuler des modifications champ par champ, sinon on pourrait alors faire apparaître des *Null* interdits et provoquer une erreur.

L'analyse de cette solution montre que nous devons impérativement stocker la mise à jour par enregistrement.

. **stockage d'enregistrement sous forme SQL**

Nous pourrions stocker la mise à jour sous forme de chaîne d'annulation SQL.

ex : Nous modifions le champ *Nom* pour l'employé dont l'id est 12

La chaîne d'annulation serait

Chaîne d'annulation SQL

```
UPDATE Employés
SET Nom = "Ancienne Valeur"
WHERE Id = 12
```

. **stockage sur le recordset**

Nous allons stocker enregistrement par enregistrement les modifications marginales sur le recordset source du formulaire.

. en cas de suppression : nous enregistrons les anciennes valeurs ;

. en cas d'ajout : nous n'enregistrons que les valeurs de clé du nouvel enregistrement, ce afin de le retrouver ou de le supprimer (annulation).

. en cas de modification : nous enregistrons les anciennes valeurs uniquement pour les changements.

Dans la pratique nous allons constituer une chaîne de mise à jour :

. un mot clé sur 6 caractères : UPDATE / DELETE / INSERT

. une liste de valeurs : la position ordinaire du champ dans le recordset suivi du signe égal et de sa valeur.

Nous avons choisi cette dernière méthode, nous allons créer une chaîne d'annulation qui opérera sur le recordset du formulaire.

id	author	type	revertAction	revertUnique	form
34	sysOP	UPDATE	UPDATE1="Davolio"	UNIQUE0=1	Employés
35	sysOP	UPDATE	UPDATE1="Davollomente", 3="Représentant(e)"	UNIQUE0=1	Employés

Les modifications seront stockées dans une table, afin de pouvoir les consulter et de laisser le choix à l'utilisateur de les annuler.

4. Table des modifications



Il s'agit de la table qui va recevoir les mises à jour faites sur le formulaire.

Nous avons besoin de recueillir les informations suivantes :

- . **login** de la personne qui modifie [**author**] ;
- . **date / heure** la mise à jour [**datUpdate**] ;
- . **type** de la modification : suppression, édition, ajout [**type**] ;
- . **nom du formulaire** sur lequel a eu lieu la modification [**form**] ;
- . **chaîne d'annulation** qui va permettre de rétablir les anciennes valeurs [**revertAction**] ;

. **chaîne unique** c'est ce qui va permettre d'identifier de manière unique l'enregistrement modifié (*nous reviendrons sur cette notion plus bas*) [**revertUnique**] .

Voici la fonction qui nous permet de créer la table.

Procédure de création de la table

```
Private Sub CreateTableLogUpdates()  
''' procédure de création de la table tblLogUpdates  
DoCmd.RunSQL "CREATE TABLE tblLogUpdates (id COUNTER, author  
TEXT, type TEXT, " &  
"revertAction MEMO, revertUnique TEXT, form  
TEXT, lot TEXT, " &  
"Field TEXT, datUpdate DATE, selection  
YESNO);"  
End Sub
```

revertAction est notre chaîne d'annulation, son type est MEMO car il faut un important stockage dans le cas où un champ de type Mémo serait modifié, la limite du type String de 255 caractères serait trop vite atteinte.

Voici la fonction qui permet de recueillir le login de la personne qui effectue la mise à jour.

Fonction qui retourne le login de la session

```
.....  
'  
' API advapi32.dll  
'  
.....  
' récupérer le login Windows  
Private Declare Function apiGetUserName Lib "advapi32.dll" Alias  
"GetUserNameA" (ByVal lpBuffer As String, nSize As Long)  
As Long  
  
Private Function GetLogin() As String  
' Retourne le nom d'utilisateur fourni lors du branchement au réseau.  
Dim lngLen As Long, lngX As Long  
Dim strUserName As String  
strUserName = String$(254, 0)  
lngLen = 255  
lngX = apiGetUserName(strUserName, lngLen)  
If lngX <> 0 Then  
GetLogin = Left$(strUserName, lngLen - 1)  
Else  
GetLogin = ""  
End If  
  
End Function
```

Retrouvez la suite de l'article de Charles A. en ligne [[Lien83](#)]

Vu dans la FAQ




[Comment lister les processus actifs sans utiliser l'API Windows \(Win2000 et >\) ?](#)

Il est possible de lister les processus actifs sur une machine en utilisant les fonctions de l'API Windows

Mais il existe une astuce beaucoup plus simple à l'aide d'un petit script WMI, utilisant la classe Win32_process (testé sur Windows 2000, XP)

vb

```
Private Sub Command1_Click()  
Dim svc As Object  
Dim sQuery As String  
Dim oproc  
On Error GoTo Command1_Click_Error  
Set svc = GetObject("winmgmts:root\cimv2")  
sQuery = "select * from win32_process"  
For Each oproc In svc.execquery(sQuery)  
Debug.Print oproc.Name & " = " & oproc.ExecutablePath  
Next  
Set svc = Nothing  
Exit Sub  
Command1_Click_Error:  
MsgBox "Error " & Err.Number & " (" & Err.Description & ")"  
Err.Clear  
End Sub
```

Retrouvez toutes les informations sur la classe Win32_process (propriétés et méthodes) sur le site de Microsoft MSDN :  [Win32_Process WMI class](#)

Attention, certaines propriétés ne sont pas implémentées. Lire attentivement les descriptions et explications fournies sur le site !

Retrouvez l'ensemble de la FAQ VB en ligne [[Lien84](#)]

Les derniers tutoriels et articles

Rédaction des documents d'analyse avec U.M.L

Cet article décrit la démarche d'écriture des documents de spécifications en s'appuyant sur la notation UML. Il part de la réponse au cahier des charges jusqu'à la rédaction du dossier de conception. Il ne prétend pas être une analyse ou un document méthodologique, simplement un exemple essayant de donner la réponse à une question que nous rencontrons souvent lors des formations « Quand dois-je utiliser tel ou tel schéma ? ». Nous avons préféré mettre en avant les commentaires de la démarche plutôt que la rédaction d'un document de spécification en tant que tel et nous avons privilégié une lecture transversale du document vous permettant de comprendre rapidement comment on s'appuie sur les schémas pour la rédaction. Le cas exposé est décrit dans le cahier des charges. Il est, comme tout cahier des charges, insuffisant, ce qui introduit la notion de risque inhérente à tout projet. La démarche de construction va donc s'enrichir dans les phases ultérieures de réunions dont nous donnons le résumé par la rédaction de (fictifs) comptes-rendus. Il ne s'agit pas d'un cours UML, ni d'un plan qualité mais seulement d'une approche vous permettant, nous l'espérons, de visualiser l'apport d'UML dans une rédaction. Bonne lecture ...

1. L'auteur

Pierre Meleard est chef de projet.

Depuis 1998, Pierre Meleard a dirigé des projets au forfait qui l'on amené à s'intéresser au design et à l'eXtreme Programming. Il a mené de nombreuses missions comme consultant (Gestion de projet, intégration des nouvelles technologies de développement, ...) ou comme formateur (Software design, Design pattern, tests unitaires, Swing, J2EE, ...). En 2005, Pierre Meleard a intégré le cabinet Infhotep comme directeur technique.

2. Cahier des charges

2.1. Préambule

Vos commerciaux font bien leur travail. Votre société est identifiée comme acteur principal des communautés de commerçants et vos solutions sont connues comme innovantes et techniquement à la pointe du mouvement Vous recevez donc le cahier des charges que voici : (Non je sais, un cahier des charges comme celui-ci, vous en rêvez, disons qu'il est le compte-rendu de réunions préliminaires ...)

2.2. Le besoin

Des commerçants souhaitent participer à l'animation du centre ville par la création d'un groupement de commerçants dont les principaux objectifs sont :

- Attirer la clientèle en centre ville par la création d'une carte de fidélité
- Animer des opérations de marketing (Semaine de promotion, ...)
- Constituer un carnet d'adresse des clients pour des opérations de mailing ciblés.

Le modèle économique adopté est le suivant :

- Tous les mois, le groupement demande aux commerçants adhérents de verser une cotisation de fonctionnement
- Sur chaque achat donnant lieu à un acte de fidélisation, le commerçant accorde une remise de 5 % sur le montant de l'achat. Sur cette remise, est prélevée une participation au

frais destiné au prestataire informatique.

- Le budget initial de mise en place du groupement relève de subventions ou de sponsoring.

Le groupement n'ayant que peu de moyens financiers souhaite des propositions d'un prestataire informatique pour la mise en place de la carte de fidélité et son hébergement.

Le prestataire pourra se rémunérer par des frais fixes pris sur chacun des actes de fidélisation (montant prélevé sur la remise accordée).

2.2.1. Architecture

Tous les commerçants sont dotés d'un TPE (Terminal de Paiement Electronique) sur lequel, à coté du logiciel de paiement bancaire, peut être installé un logiciel de fidélité.

Le TPE dialogue avec un serveur hébergé chez le prestataire qui mémorise les transactions et effectue les calculs de fidélité. Le serveur doit disposer d'une application permettant d'effectuer la comptabilité de fin de mois en liaison avec les banques et de calculer les statistiques.

Un serveur WEB doit permettre au groupement de saisir les informations nécessaires au fonctionnement de la fidélité.

2.2.2. Principe

Le principe de la fidélité est le suivant :

- Chaque client est possesseur d'une carte de fidélité magnétique valide dans tous les commerçants du groupement.
- Chaque achat donne lieu à une provision de remise (Montant d'achat * 5% - frais). Le commerçant est débité chaque mois des provisions accordées plus la cotisation de fonctionnement au profit d'un compte détenu par le groupement.
- A bout du dixième achat, la somme des provisions (moins les frais) devient une remise accordée au client qui est déduite par le commerçant du montant de ve dixième achat. Si la remise accordée est supérieure au montant de l'achat, le commerçant débourse de sa caisse la différence. Le commerçant est crédité du montant de la remise accordée, en fin de mois, à partir du compte du

groupement.

2.2.3. Acte de fidélité

Lorsqu'un nouveau client arrive, le commerçant donne au client une carte magnétique au logo du groupement permettant d'effectuer les actes de fidélité. Le commerçant demande au client de remplir un coupon papier sur lequel le client donne son adresse. Le commerçant complète le coupon avec le numéro de la carte de fidélité. Le coupon est envoyé à la secrétaire du groupement qui effectue la saisie des adresses.

L'enregistrement d'un acte d'achat donnant lieu à une remise au titre de la fidélité est le suivant :

1. Le commerçant saisit le montant de l'achat
2. Le commerçant introduit la carte de fidélité
3. Le TPE appelle un serveur chez le prestataire informatique par le réseau commuté
4. Le serveur reçoit la demande de fidélité avec le numéro du commerçant, le numéro de la carte et le montant. Il vérifie l'existence de la carte, du commerçant et calcule le montant de la fidélité.
5. 1er cas :
 - Si le client n'a pas droit à la remise, le serveur renvoie la provision de remise accordée
6. Le TPE édite un ticket en double exemplaire (l'un pour le

commerçant, l'autre pour le client) avec le nom du commerçant, la date d'achat, le montant de l'achat, la provision de remise, le cumul des provisions et le rang de l'achat

7. Le commerçant encaisse la totalité du montant de l'achat
8. 2ème cas :
 - Le client a droit à une remise, le serveur renvoie les informations permettant le remboursement de la remise (somme des provisions - somme des frais)
9. Le TPE édite un ticket comprenant, le nom du commerçant, la date d'achat, la provision de remise sur cet achat, le montant des provisions de remise, et le restant dû par le client (Achats - remise). Il est possible que la somme des remises soit supérieure au montant de l'achat.
10. Le commerçant encaisse le montant de l'achat moins la remise ou décaisse l'écart.

2.2.4. Site Web

Le site Web doit permettre au groupement de paramétrer l'application sur le serveur, de saisir les adresses et d'effectuer des interrogations sur la vie des cartes. Peu de commerçants disposent d'internet. La secrétaire du groupement est la principale utilisatrice du site Web.

Retrouvez la suite de l'article de Pierre Meleard en ligne [[Lien85](#)]

Livres



Le projet d'urbanisation du S.I.

Démarche pratique avec cas concret
284 pages, 2ème édition, 15 Janvier 2004 Editions
Dunod, ISBN: 2100073761
Public visé : **Intermédiaire**
Commandez sur [Amazon.fr](#)

La notion d'urbanisme se situe bien évidemment dans le prolongement de celle d'architecture des systèmes d'information. Cet ouvrage montre comment organiser le S.I. d'une entreprise de façon suffisamment modulaire pour pouvoir rénover une fonction par exemple la gestion des stocks sans paralyser l'ensemble de l'entreprise tout en définissant les principes et les protocoles permanents qui assureront la cohérence et le fonctionnement de l'ensemble sur le long terme. Les principes de l'urbanisation ainsi que la démarche méthodologique sont illustrés dans ce livre par la mise en oeuvre d'un cas concret.
Public concerné : Directeurs des S.I - Directeurs informatiques - Maîtres d'ouvrages - Consultants.

Christophe Longépé est directeur de l'urbanisme, de l'architecture fonctionnelle et des référentiels à la Société Générale et chargé de cours à VIMI (Institut du Management de l'Information - Université de Technologie de Compiègne). Il était précédemment directeur associé au sein de SchlumbergerSema Consulting en charge de la pratique " Stratégie, urbanisme et architecture des systèmes d'information ".

Critique du livre par la rédaction ([Pierre Chauvin](#)) :

Un **projet d'urbanisation d'un système d'information (S.I)** est

une tâche complexe. Trouver une adhésion de tous les membres de l'organisation n'est pas aisé, mais cela fait partie du jeu: vous devez montrer les besoins réels d'une urbanisation, les lacunes du système existant, et ainsi construire un dialogue impliqué et aboutissant sur un projet de bénéfice global pour l'entreprise.

C'est cette démarche méthodologique que *Christophe Longépé* vous propose. Il explique de quelle manière on peut structurer un tel projet, d'un point de vue de l'organisation et de planification de l'implémentation.

Le livre est bien structuré, la terminologie utilisée est claire et expliquée. Sans entrer dans une critique de fond que ne peux objectivement fournir, les différents chapitres m'ont appris un certain nombre de choses: par où doit-on commencer ? qui doit travailler sur un projet d'urbanisation ? quel dialogue doit on avoir ? quelles étapes faut-il respecter ?. On comprendra donc davantage les **objectifs de l'urbanisme**, son périmètre et son intérêt pour les différents composants d'une entreprise, ou encore **comment procéder à une cartographie efficace de l'existant**, quelles **bonnes pratiques** sont à adopter afin de maîtriser les **processus métier** et de trouver une convergence avec la stratégie et l'architecture déjà mise en place.

Cette seconde édition mérite beaucoup d'attention. Sans y trouver une "Bible", vous y puiserez certainement de riches idées et des explications de qualité. Je conseille cet ouvrage aux (futurs?) urbanistes, architectes, DSI, chef de projets, ingénieurs qualité.

[Consulter la critique et le descriptif complet...\[Lien86\]](#)



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.

- Lien1 : <http://tahe.developpez.com/java/springmvc-part1/>
- Lien2 : <http://tahe.developpez.com/#L1>
- Lien3 : <http://brabant.developpez.com/javapolis/2005/>
- Lien4 : <http://blog.developpez.com/index.php?blog=39&p=1892&more=1&c=1&tb=1&pb=1>
- Lien5 : <http://fr.sun.com/>
- Lien6 : http://blog.developpez.com/index.php?blog=12&title=codcamp_2006_dans_votre_salon&more=1&c=1&tb=1&pb=1
- Lien7 : <http://java.developpez.tv/codcamp2006/>
- Lien8 : <http://merlin.developpez.com/cours/dotnet/WeakReference/>
- Lien9 : <http://ditch.developpez.com/aspnet2/membershipprovider/>
- Lien10 : <http://www.aisto.com/roeder/dotnet/>
- Lien11 : <http://lgorand.developpez.com/dotnet/smartassembly/>
- Lien12 : http://blog.developpez.com/index.php?blog=9&title=date_de_sortie_des_services_pack_de_visu&more=1&c=1&tb=1&pb=1
- Lien14 : <http://blogs.msdn.com/dannawi/archive/2006/04/27/585311.aspx>
- Lien15 : http://blog.developpez.com/index.php?blog=9&title=precisions_sur_les_dates_de_sortie_des_s&more=1&c=1&tb=1&pb=1
- Lien16 : <http://www.developpez.net/forums/member.php?u=3901>
- Lien17 : http://blog.developpez.com/index.php?blog=17&title=les_devdays_2006_sur_developpez_tv&more=1&c=1&tb=1&pb=1
- Lien18 : <http://dotnet.developpez.tv/devdays2006/>
- Lien19 : <http://taiani.developpez.com/tolerance-faute/>
- Lien20 : <http://jacquenod.developpez.com/cpl-courant-porteur-en-ligne/>
- Lien21 : <http://mtodorovic.developpez.com/linux/bootsplash/>
- Lien22 : <http://dico.developpez.com/html/1542-Telecom-WiMAX-Worldwide-Interoperability-for-Microwave-Access.php>
- Lien23 : http://www.onduleurs.fr/infocenter.php?icPath=7_10
- Lien24 : http://www.onduleurs.fr/infocenter.php?icPath=7_21
- Lien35 : <http://www.debian.org/>
- Lien26 : http://www.apc.com/resource/include/techspec_index.cfm?base_sku=BR8001
- Lien27 : <http://www.mgeups.com/products/pdt230/smallups/cl/cl.htm>
- Lien28 : <http://linux.developpez.com/cours/upsusb/>
- Lien29 : <http://bwp-necromance.developpez.com/tutoriel/linuxmandriva2006/>
- Lien30 : <http://c.developpez.com/livres/#L2841772896>
- Lien31 : http://blog.developpez.com/index.php?blog=79&title=prise_de_controle_a_distance_sur_un_pc_n&more=1&c=1&tb=1&pb=1
- Lien32 : <http://linux.developpez.tv/solutionslinux2006/>
- Lien33 : <http://merlin.developpez.com/cours/delphi/dotnet/portage1/>
- Lien34 : <http://merlin.developpez.com/cours/delphi/dotnet/portage2/>
- Lien35 : <http://thierryaim.developpez.com/tutoriel/adodelphiperso/>
- Lien36 : http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/hdm/odbcodbc_api_reference.asp
- Lien37 : <http://msdn.microsoft.com/data/mdac/downloads/default.asp>
- Lien38 : http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/hdm/odbcabout_this_manual.asp
- Lien39 : http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/hdm/odbcodbc_api_reference.asp
- Lien40 : <http://trafalga.free.fr/iodbc.zip>
- Lien41 : <http://delphi.developpez.com/faq/>
- Lien42 : <http://delphi.developpez.com/faq/?page=odbc#creationodbc>
- Lien43 : <http://okaryn.developpez.com/articles/delphi/iodbc/>
- Lien44 : <http://gfx.developpez.com/java/wicket/>
- Lien45 : <http://www.amazon.fr/exec/obidos/ASIN/221211768X/wwwdeveloppez-21>
- Lien46 : <http://laurent-dardenne.developpez.com/articles/Delphi/RTTI/>
- Lien47 : <http://noiseteprod.developpez.com/delphi/doc/pdf7/>
- Lien48 : <http://www.developpez.net/forums/showthread.php?t=105012>
- Lien49 : <http://siddh.developpez.com/articles/ajax/>
- Lien50 : <http://www.google.com/webhp?complete=1&hl=en>
- Lien51 : <http://dcabasson.developpez.com/articles/javascript/ajax/ajax-autocompletion-pas-a-pas/fichiers/options.php?debut=de>
- Lien52 : <http://siddh.developpez.com/articles/ajax/>
- Lien53 : <http://dcabasson.developpez.com/articles/javascript/ajax/ajax-autocompletion-pas-a-pas/>
- Lien54 : <http://sznajderman.developpez.com/articles/ergonomie/>
- Lien55 : http://blog.developpez.com/index.php?blog=51&title=navigateur_et_standards_web&more=1&c=1&tb=1&pb=1
- Lien56 : http://blog.developpez.com/index.php?blog=30/index.php?blog=30&title=comment_installer_ror_manuellement&more=1&c=1&tb=1&pb=1
- Lien57 : <http://xmlrad.developpez.com/>
- Lien58 : <http://rdm.developpez.com/XMLRAD/>
- Lien59 : http://xmlrad.com/DelosBin/Delos.dll/ServePage?URL=Manual/Appendix14/index.htm&WEB_ID=101001015
- Lien60 : http://xmlrad.com/DelosBin/Delos.dll/ServePage?URL=Manual/Appendix03/index.htm&WEB_ID=101001015
- Lien61 : http://xmlrad.com/DelosBin/Delos.dll/ServePage?URL=Manual/Appendix09/index.htm&WEB_ID=101001015
- Lien62 : <http://xmlrad.developpez.com/Articles/ApplicationPHP/>
- Lien63 : <http://www.developpez.net/forums/member.php?u=52698>
- Lien64 : <http://oracle.developpez.com/guide/tuning/tkprof/>
- Lien65 : http://download-east.oracle.com/docs/cd/B19306_01/server.102/b14211/sqltrace.htm#i4191
- Lien66 : http://download-east.oracle.com/docs/cd/B19306_01/server.102/b14237/initparams204.htm#REFRN10208
- Lien67 : http://download-east.oracle.com/docs/cd/B19306_01/server.102/b14237/initparams065.htm#REFRN10054
- Lien68 : <http://leoanderson.developpez.com/traces-sessions-oracle/>
- Lien69 : <http://leoanderson.developpez.com/securisation-oracle/>
- Lien70 : <http://support.microsoft.com/default.aspx?scid=kb;fr;278379>
- Lien71 : <http://arkham46.developpez.com/articles/access/mousewheel/>
- Lien72 : <http://loufab.developpez.com/recursivite2/>
- Lien73 : <http://4d.developpez.com/ajax/>
- Lien74 : <http://webster.cs.ucr.edu/>
- Lien75 : <http://www.alapage.com/-/Fiche/Livres/2744070025/?id=169491135333860>
- Lien76 : <http://man.developpez.com/man3/fgets.3.php>
- Lien77 : <http://man.developpez.com/man3/strncpy.3.php>
- Lien78 : <http://man.developpez.com/man3/strncat.3.php>
- Lien79 : <http://www.grsecurity.net/>
- Lien80 : <http://c.developpez.com/equipe/>
- Lien81 : <http://nicolasj.developpez.com/articles/pieges/>
- Lien82 : <http://sql.developpez.com/sqlserver/transactsql/#L5>
- Lien83 : <http://cafeine.developpez.com/access/tutoriel/update/>
- Lien84 : <http://vb.developpez.com/faq/>
- Lien85 : <http://meleard.developpez.com/uml/redaction/document/>
- Lien86 : <http://wpetrus.developpez.com/livres/urbanisation/>