

Manuel utilisateur pour Tulip IDE

Table des matières

Abstract.....	1
Introduction.....	1
Gratuit.....	2
PHP-GTK.....	2
Libre.....	2
L'interface.....	2
Les menus.....	3
Menu Fichier.....	3
Menu Édition.....	4
Menu CVS.....	5
Menu d'aide.....	5
Les boutons.....	6
La zone de texte.....	6
Onglets.....	6
Marge.....	7
Zone d'édition.....	7
La zone de hiérarchie des fichiers.....	7
Les projets.....	8
Les projets, à quoi bon?.....	8
Ouvrir un nouveau projet.....	8
Changer de projet.....	9
Enregistrer un projet.....	9
Fermer un projet.....	9
CVS.....	9
Paramétrer CVS.....	9
Bases de données.....	9
Connexion à une base de données.....	9
Interface de gestion.....	10
FAQ.....	10

Abstract

Ce manuel se réfère à la version CVS équivalente à la version 0.3 officielle de *Tulip* disponible le 2004-08-09. Ce manuel n'a pas (encore) de licence par méconnaissance de l'auteur des types de licences disponible. Vous pouvez en faire ce que vous voulez tant que vous conservez les informations contenues dans cet abstract. Pour toute suggestion, n'hésitez pas à nous envoyer un mail sur tulip-english@codigolivre.org.br (pour s'inscrire, c'est ici: <http://listas.codigolivre.org.br/mailman/listinfo/tulip-english>)

Version de <i>Tulip</i> :	0.3
Développeur principal:	Pablo Dall'Oglio
Sponsors:	Solis- Free software cooperative
Auteur de ce document:	Yannick Warnier
Date:	2004-08-09
Origine:	https://codigolivre.org.br/docman/?group_id=739

Introduction

Tulip est un IDE (*Integrated Development Environment*), c'est-à-dire un environnement de développement intégré. Intégré dans quoi? Dans votre système.

C'est, au même titre que le produit vendu par *Zend* (société très active dans le développement de *PHP*) et certaines extensions d'*Eclipse* et d'autres produits encore, un outil très utile quand on en vient à la programmation en *PHP*. Utile mais surtout rapide, grâce à sa fonction d'auto-complétion des fonctions. Elle vous permettra un gain de temps important car vous ne devrez plus consulter la documentation de *PHP* que très rarement.

Mais *Tulip* offre aussi trois avantages non négligeables

- il est gratuit
- il est écrit en *PHP-GTK*
- il est libre

Qu'est-ce que tout cela peut bien vouloir dire? Voyons plus en détail ces trois avantages...

Gratuit

Tulip est gratuit. Ce qui veut dire que vous pouvez le charger, l'essayer, avec sa fonctionnalité complète, et décidez si vous désirez le garder ou pas. Vous n'en voulez pas? Désinstallez-le. Vous voulez le garder? Gardez-le donc! C'est à vous de choisir, sans déboursier un centime, quelque soit votre choix.

Comment cela se fait-il? *Tulip* est développé par des développeurs *PHP*, tout comme vous. Par des gens qui ont besoin de cet outil précieux. Ils n'ont pas d'intérêt à le vendre car cela demanderait des compétences qu'ils n'ont pas et beaucoup de travail pour un gain probablement minime.

Quoi qu'il en soit, la question ne se pose pas. L'outil est développé par ceux qui en ont besoin. Vous pouvez d'ailleurs nous aider, soit en programmant avec nous, soit en nous envoyant des rapports d'erreurs, si vous en trouvez. Pour nous envoyer vos erreurs, utilisez l'interface de *codigolivre* ici: https://codigolivre.org.br/bugs/?group_id=739 afin de le faire en toute simplicité et de façon à ce que nous puissions les traiter au plus vite.

PHP-GTK

Le fait que cette application soit écrite en *PHP-GTK*, vous permet à vous, développeur *PHP*, de facilement modifier l'application afin de la faire correspondre à vos besoins. Beaucoup d'entre vous ne profiteront pas de cette possibilité, mais si un jour vous en avez besoin, vous comprendrez à quel point c'est important et bénéfique.

Libre

Et puis après tout, le plus important est que la modification du code soit possible, légalement et en toute simplicité. Le fait que le code source de l'application soit libre vous permet d'en faire ce que vous voulez. Seules quelques règles s'appliquent. Ces règles se retrouvent dans la licence de l'application, dont une référence est notamment disponible dans le fichier de démarrage de l'application: *tulip.php*.

Non seulement il vous est possible de modifier le code parce qu'il est écrit en *PHP*, mais en plus, c'est légal!

L'interface

Lors du premier démarrage de *Tulip*, vous verrez un écran similaire à celui repris en figure 1. Il s'agit d'un écran tout-à-fait commun pour un environnement de développement. C'est dans cet environnement que vous resterez tout au long de votre expérience de *Tulip*.

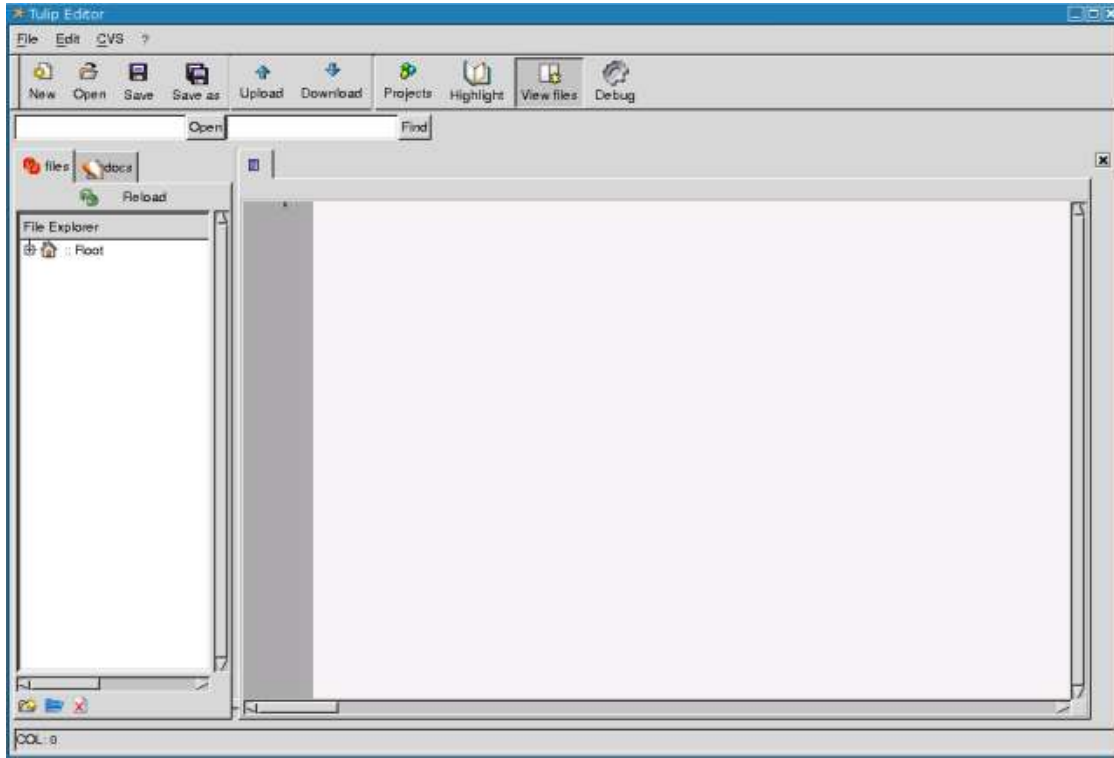


Illustration 1 - Écran principal

Comme vous pouvez vous même le constater, l'interface se découpe en 5 parties. Dans l'ordre du haut à gauche vers le bas à droite, nous trouvons respectivement les *menus*, la *barre d'outils*, la *barre de recherche*, l'*arborescence des fichiers* et enfin la partie *édition*.

Il reste encore une dernière partie tout en dessous, que nous ne verrons pas en détail ici mais que nous pouvons citer rapidement, il s'agit de la *barre d'état*, dans laquelle nous verrons parfois apparaître des messages contextuels explicatifs.

Les menus

Menu Fichier

Celui-ci vous offre des options classiques, mais nous repassons-les rapidement en revue...

New (Nouveau) Ouvre un nouveau fichier dans la partie *édition* de l'interface.

Open (Ouvrir) Ouvre un fichier existant dans la partie *édition*.

Open Recent (Ouvrir récent)

Ouvre des fichiers récemment ouverts par *Tulip*.

Save (Enregistrer) Sauvegarde les modifications dans le fichier ouvert dans la partie *édition*.

Save as (Enregistrer sous)

Sauvegarde les modifications du fichier ouvert dans la partie *édition* sous **un autre** fichier.

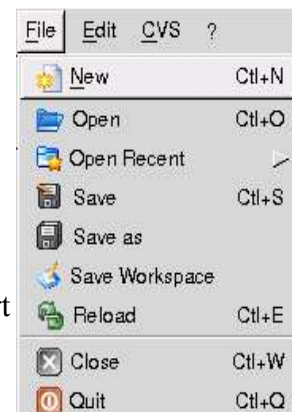


Illustration 2 - Menu Fichier

Save Workspace (Sauvegarder tout)

Sauvegarde tous les fichiers ouverts dans la partie *édition*.

Reload

Rafraîchit le contenu du fichier actuellement ouvert, afin d'obtenir une version plus récente si elle a été modifiée depuis l'extérieur.

Close

Ferme le fichier actuellement ouvert dans la partie *édition*.

Quit

Quitte *Tulip*. Cette action est associée à la sauvegarde des fichiers ouverts.

Menu Édition

Celui-ci comporte, depuis la version 0.3, de nombreuses possibilités.

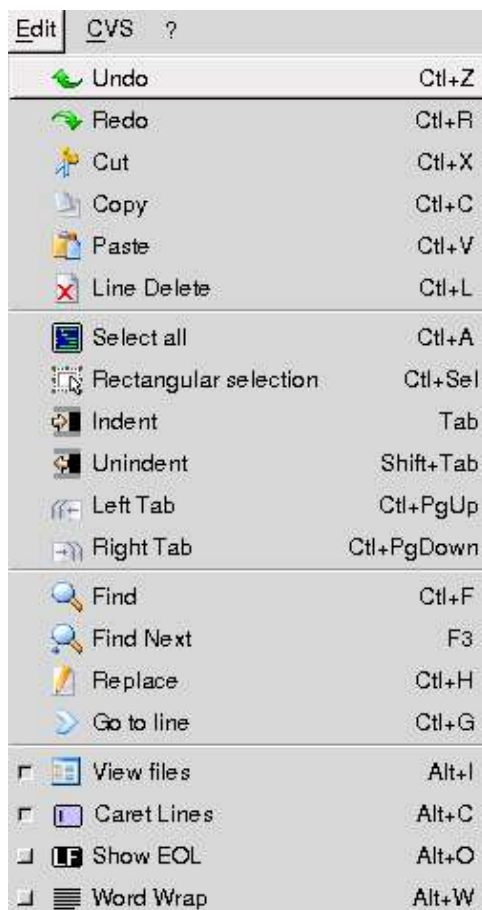


Illustration 3 - Menu Édition

Undo (Annuler)

Annule la dernière modification dans la pile des modifications enregistrées sur les fichiers ouverts dans la partie *édition*.

Redo (Restaurer)

Annule la dernière annulation.

Cut (Couper)

Coupe un morceau de texte et le place dans la mémoire tampon pour utilisation ultérieure.

Copy (Copier)

Copie un morceau de texte dans la mémoire tampon pour utilis. ultérieure.

Paste (Coller)

Colle à l'emplacement du curseur le morceau de texte gardé en mémoire.

Line Delete (Effacer une ligne)

Efface la ligne sur laquelle se trouve le curseur.

Select all (Tout sélectionner)

Sélectionne tout le texte du fichier courant.

Rectangular selection (Sélection rectangulaire)

Sélectionne un rectangle de texte défini par l'utilisateur à l'aide de la souris.

Indent (Indenter)

Déplace de quelques espaces vers la droite la sélection ou la ligne courante.

Unindent (Désindenter)

Déplace de quelques espaces vers la gauche la sélection ou la ligne active

Left tab (Tabulation à gauche)

Passé à l'onglet de gauche s'il existe dans la partie *Édition*

Right tab (Tabulation à droite)

Passé à l'onglet de droite s'il existe dans la partie *Édition*

Find (Chercher)

Ouvre une fenêtre de recherche d'un texte dans le fichier courant.

Find next (Chercher suivant)

Cherche l'occurrence suivante du terme recherché avec *Find*.

Replace (Remplacer)

Remplace toutes les occurrences d'une chaîne de caractères par une nouvelle chaîne de caractères

Go to line (Aller à la ligne)

Permet de se rendre directement à une certaine ligne.

View files (Afficher fichiers)

Active/désactive l'affichage de l'arborescence des fichiers.

Caret lines (Surligner)

Active/désactive le surlignage de la ligne active.

Show EOL (Afficher EOL)

Active/désactive l'affichage de balises indiquant les caractères de fin de ligne (End Of Line) qui seraient normalement invisibles.

Word wrap (Enrobage des mots)

Active/désactive le passage à la ligne dans la partie *édition* lorsque les mots dépassent la largeur de la zone

Menu CVS

Le menu *CVS* propose des options réservées à *CVS* (discuté plus tard dans ce manuel).

New (Nouveau)

Crée une nouvelle entrée dans le dépôt *CVS* propre au projet actuel.

Update (Mettre à jour)

Met à jour le fichier *CVS* depuis le fichier courant (correspond à un *cvs commit*).

Delete (Supprimer)

Efface le fichier actuel du dépôt *CVS*.

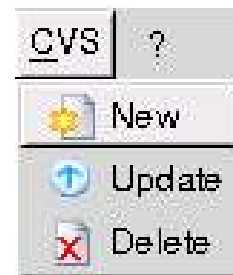


Illustration 4 - Menu CVS

Menu d'aide

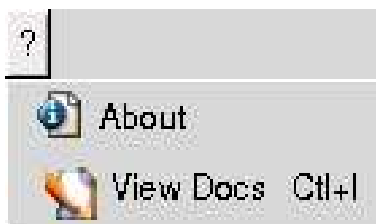


Illustration 5 - Menu d'aide

Le menu d'aide propose deux options...

About (A propos) Affiche les informations relatives au développement de *Tulip*.

View Docs (Aide) Affiche l'aide de *Tulip*.

Les boutons

La première partie de la zone de boutons est un raccourci pour les commandes des menus. Voyons le reste plus en détail.



Illustration 6 - Barre de boutons

Upload (Envoyer)

Envoie le fichier actuel sur le serveur *FTP* configuré pour le projet courant.

Download (Télécharger)

Charge un fichier depuis le serveur *FTP* configuré pour le projet courant.

Projects (Projets)

Ouvre la fenêtre de paramètres des projets.

Highlight (Mettre en valeur)

Visualise le fichier courant dans un navigateur.

View Files (Visualiser arborescence)

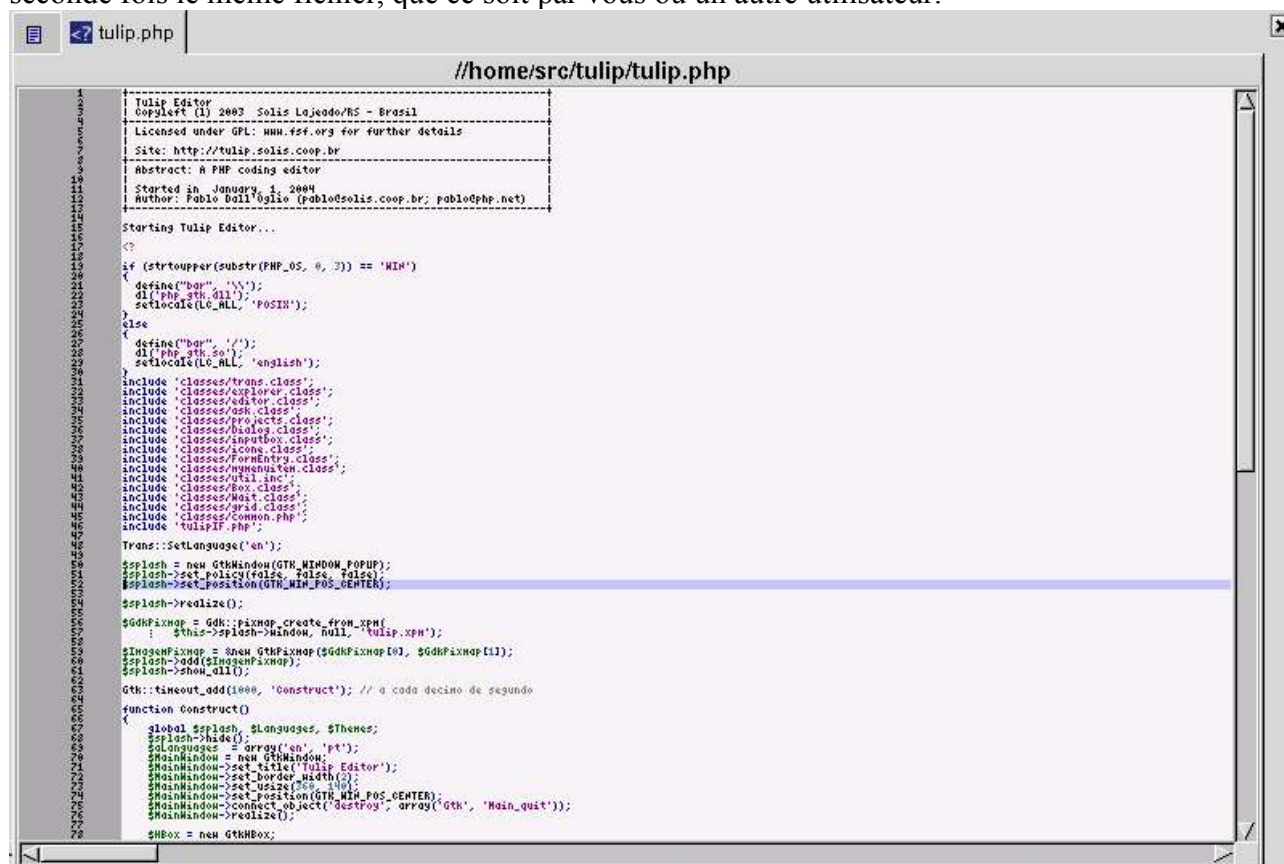
Active/désactive la zone de l'arborescence des fichiers.

Debug (Débugger)

Tente d'exécuter le fichier en ligne de commande pour repérer les fautes *PHP*.

La zone de texte

La zone de texte est celle où les fichiers divers sont affichés pour permettre de les éditer. Retenez qu'une fois qu'un fichier texte est ouvert dans *Tulip*, un nouveau fichier du même nom terminé par un ~ est créé au même niveau que le fichier original. C'est ce fichier qui empêchera d'ouvrir une seconde fois le même fichier, que ce soit par vous ou un autre utilisateur.



```
1  Tulip Editor
2  Copyright (c) 2003 Solis Lajeado/RS - Brasil
3  Licensed under GPL: www.tsf.org for further details
4  Site: http://tulip.solis.coop.br
5  Abstract: A PHP coding editor
6
7  Started in January, 1, 2004
8  Author: Pablo Dall'Oglio (pablo@solis.coop.br; pablo@php.net)
9
10 Starting Tulip Editor...
11
12
13
14
15
16
17
18
19
20
21 if (strtoupper(substr(PHP_OS, 0, 3)) == 'WIN')
22 {
23     define("bar", "\\");
24     dl('php_ssh.dll');
25     setlocale(LC_ALL, 'POSTN');
26 }
27 else
28 {
29     define("bar", '/');
30     dl('php_ssh.so');
31     setlocale(LC_ALL, 'english');
32 }
33
34 include 'classes/trans.class';
35 include 'classes/explorer.class';
36 include 'classes/editor.class';
37 include 'classes/ask.class';
38 include 'classes/projects.class';
39 include 'classes/dialog.class';
40 include 'classes/inputbox.class';
41 include 'classes/icon.class';
42 include 'classes/FormEntry.class';
43 include 'classes/MenuItems.class';
44 include 'classes/Gui.inc';
45 include 'classes/Box.class';
46 include 'classes/Wait.class';
47 include 'classes/grid.class';
48 include 'classes/Common.php';
49 include 'tulip.php';
50
51 Trans::SetLanguage('en');
52
53 $splash = new GtkWindow(GTK_WINDOW_POPUP);
54 $splash->set_policy(GTK_POLICY_ALWAYS);
55 $splash->set_position(GTK_WIN_POS_CENTER);
56
57 $splash->realize();
58
59 $gtkPixmap = Gdk::pixmap_create_from_xpml(
60     ! $this->splash->window, null, "tulip.xpm");
61
62 $img GTKPixmap = $new GtkPixmap($gtkPixmap[0], $gtkPixmap[1]);
63 $splash->add($img GTKPixmap);
64 $splash->show_all();
65
66 Gtk::timeout_add(1000, 'construct'); // a cada decimo de segundo
67
68 function construct()
69 {
70     global $splash, $languages, $themes;
71     $splash->hide();
72     $languages = array('en', 'pt');
73     $mainWindow = new GtkWindow;
74     $mainWindow->set_title('Tulip Editor');
75     $mainWindow->set_border_width(2);
76     $mainWindow->set_usize_hints(Gtk_USIZE_HINTS_NONE);
77     $mainWindow->set_position(GTK_WIN_POS_CENTER);
78     $mainWindow->connect_object('destroy', array($this, 'Main_quit'));
79     $mainWindow->realize();
80     $hBox = new GtkHBox;
```

Illustration 7 - Zone d'édition

On distingue plusieurs parties utiles, en excluant les barres de défilement, dont l'usage est évident.

Onglets

Les onglets définissent l'ensemble des fichiers ouverts actuellement. Lorsqu'un fichier a été modifié depuis son ouverture, il est précédé du symbole *, afin de rappeler qu'il doit être sauvé. La croix à l'extrémité supérieure droite permet de fermer le fichier (l'onglet) qui a actuellement le focus.

Sous la barre d'onglets, on aperçoit également, en grand, le nom du fichier ouvert (ainsi que son chemin complet).

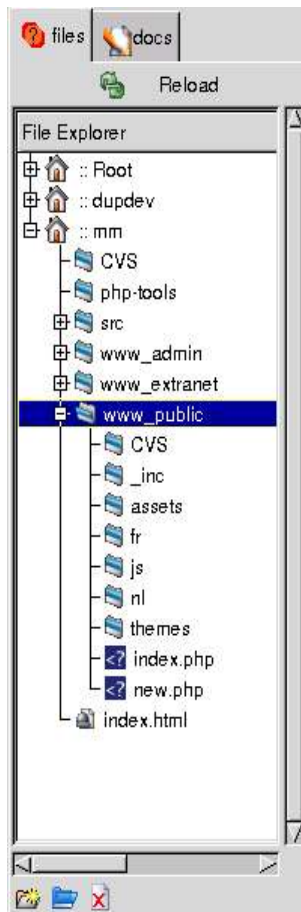
Marge

La marge indique le numéro de la ligne.

Zone d'édition

La zone d'édition est celle dans laquelle on peut modifier le texte. Notez ici la coloration syntaxique due à l'interprétation du code *PHP*, ainsi que la ligne en surbrillance, qui indique sur quelle ligne se trouve le curseur.

La zone de hiérarchie des fichiers



Enfin, la zone de hiérarchie des fichiers (à gauche quand elle est affichée) permet de se retrouver très rapidement dans les fichiers.

Elle utilise la définition des projets pour afficher des arborescences parallèles. Vous aurez donc une arborescence partant de *Root*, qui est la racine de votre système (/ sous UNIX/Linux et la plupart du temps *C:* sous *Microsoft Windows*), et d'autres arborescences dont la racine est le répertoire contenant le projet que vous avez défini. Le nom de ces arborescences parallèles est le même que celui du projet.

Lorsque vous cliquez sur un fichier dans la partie arborescences, celui-ci s'ouvre dans la partie *édition* sur la droite (et ouvre donc un nouvel onglet). Un bouton *reload* au-dessus de la zone de l'arborescence permet de rafraîchir la liste des fichiers.

Les boutons à l'extrémité inférieure permettent respectivement de créer un nouveau répertoire, d'ouvrir un répertoire (option relativement inutile dans ce cas-ci) et d'effacer un fichier ou un répertoire. Une fenêtre de confirmation s'assurera qu'il ne s'agit pas d'une erreur avant d'effacer le fichier en question.

Illustration 8 - Arborescence des fichiers

Les projets

Les projets, à quoi bon?

À quoi peut bien servir un projet? Pourquoi ne pas directement éditer les fichiers dans l'arborescence de base?

Il existe plusieurs réponses.

Premièrement, un projet permet d'isoler un certain nombre de fichiers et de les rendre plus facilement accessibles dans l'arborescence des fichiers. Mais cette première raison est de loin la moins bonne.

Deuxièmement, et c'est le plus grand avantage, un projet permet de définir, pour un certain répertoire (c'est l'unité minimum d'un projet dans l'état actuel de *Tulip*), une connexion à un serveur *FTP*, à un serveur de base de données et/ou à un serveur *CVS*. Ces diverses connexions pré-définies vous permettront d'utiliser les boutons de raccourcis pour synchroniser vos fichiers ou consulter une base de données.

Ouvrir un nouveau projet

Pour ouvrir un nouveau projet, cliquez sur le bouton *Projects (Projets)* dans la barre d'outils. Une nouvelle fenêtre apparaît. Remplissez-y les champs qui vous intéressent (au moins le nom et le répertoire local du projet) et cliquez sur *Save (Enregistrer)*.

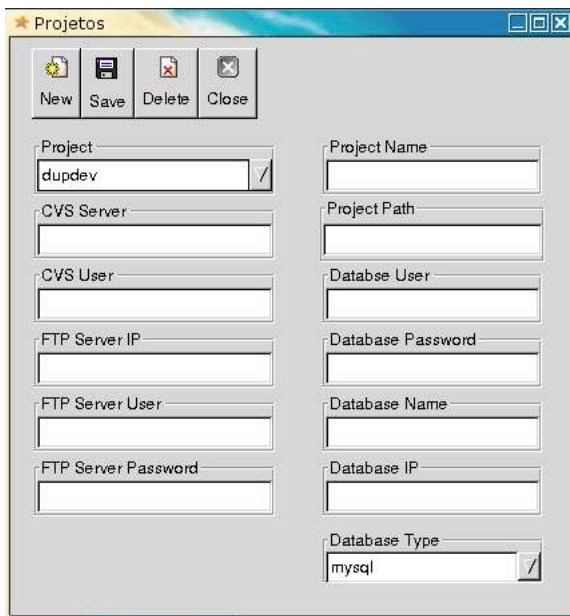


Illustration 9 - Fenêtre des projets

Fermez la fenêtre et retournez voir la barre d'arborescence des fichiers... vous avez un nouveau projet!

Changer de projet

Pour changer de projet courant, il suffit de cliquer sur le nouveau projet dans l'arborescence, ou d'en déclarer un nouveau et de le sélectionner.

Enregistrer un projet

Pour enregistrer un projet (sauvegarder l'entièreté des modifications sur ses fichiers), il n'y a pas véritablement de solution actuellement. Il faut sauvegarder chaque fichier avant de le fermer et/ou sauvegarder en un coup l'ensemble des fichiers ouverts avec l'option *Save Workspace* du menu *File*.

Fermer un projet

Pour fermer un projet, changez simplement de projet actif. Pour le supprimer de la configuration, ouvrez la fenêtre *Projects*, sélectionnez le projet à supprimer et cliquez sur *Delete (Supprimer)*.

CVS

CVS est un outil de gestion des versions d'une application. Il permet notamment de développer à plusieurs personnes en gardant un historique des modifications qui ont été faites. Si vous ne connaissez pas, il est nécessaire de lire la documentation pour comprendre son principe de fonctionnement. Vous avez également besoin d'un serveur (actuellement il doit être configuré en *pserv* pour fonctionner avec *Tulip*). Pour plus d'informations, consultez <http://www.gnu.org/software/cvs/manual/> en anglais malheureusement.

Paramétrer CVS

Paramétrer *CVS* se fait au niveau *projet*. Autrement dit, vous devez avoir un projet actif pour pouvoir y configurer *CVS*. Dans la fenêtre *Projects (Projets)*, remplissez simplement les divers champs nécessaires (adresse du serveur, identifiant, mot de passe). Vous devriez pouvoir utiliser directement les options offertes par le menu *CVS*.

À ce jour, vous ne pouvez pas encore effectuer toutes les commandes possibles avec *CVS*.

Bases de données

Les bases de données constituent une partie importante des sites web dynamiques aujourd'hui. Pour faciliter le développement web en parallèle avec une base de données, *Tulip* offre la possibilité de configurer une connexion à une base de données via l'interface des projets. Seuls deux systèmes de gestion de bases de données sont actuellement gérés par *Tulip*. Ils sont tous les deux libres et gratuits (pour *MySQL* il y a toutefois des limitations à ces liberté et gratuité), il s'agit de *PostgreSQL* et de *MySQL*. Si *PostgreSQL* a beaucoup plus l'envergure d'une base de données

professionnelle, *MySQL* est quant à lui beaucoup plus populaire et les deux systèmes ont leurs qualités et leurs défauts.

Tulip vous offre donc la possibilité de configurer une connexion à une base de données (par projet), et d'ensuite consulter les tables de cette base de données à partir de *Tulip* même. Les possibilités techniques sont nombreuses grâce à la possibilité d'exécuter des commandes *SQL* directement sur les tables.

Connexion à une base de données

La connexion à une base de données se fait au moment de l'activation du projet. La base de données se retrouve alors dans la hiérarchie des fichiers du projet, comme s'il s'agissait d'un répertoire. Cliquer dessus déploiera la liste des tables de la base.

Interface de gestion

Comme le montre l'illustration suivante, l'interface de la base de données est très flexible et permet une utilisation approfondie de celle-ci.

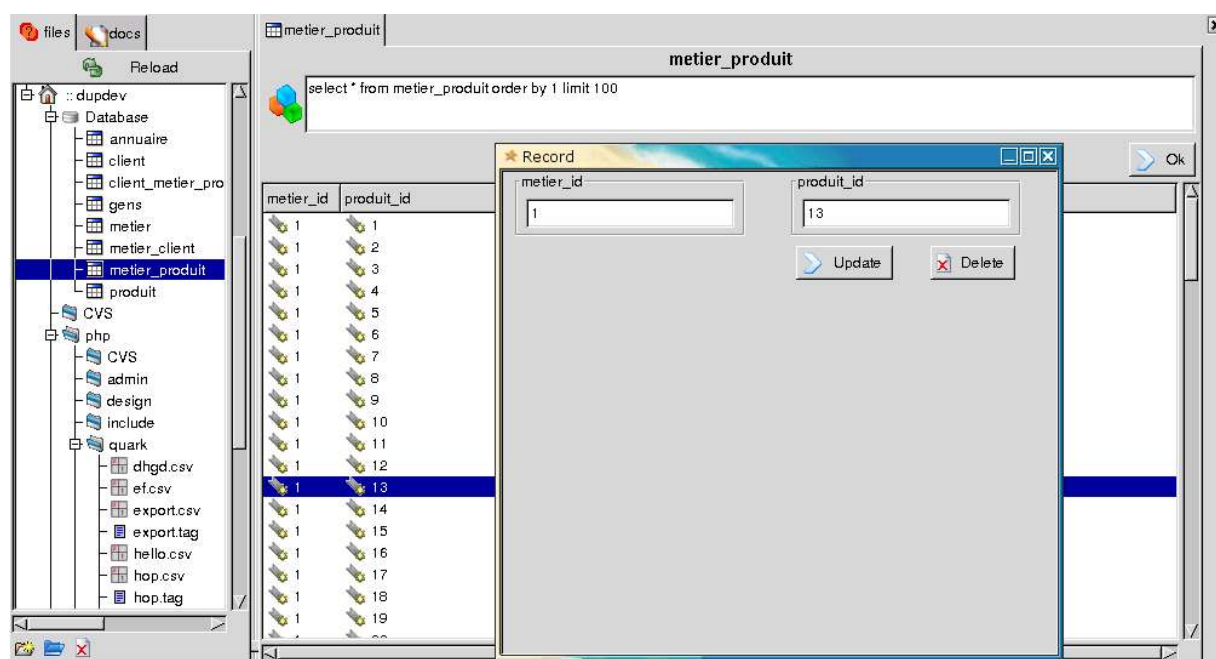


Illustration 10 - Fenêtre de base de données

Lorsque l'on clique sur une table, les détails de la table s'affichent dans un nouvel onglet dans la partie *Édition*. On peut directement y lancer une requête *SQL*.

Si l'on clique sur une ligne de la table, une nouvelle fenêtre s'ouvre avec des détails sur la rangée sélectionnée, et l'on peut directement modifier les valeurs de la rangée, ou la supprimer.

Les changements sont réellement opérés dans la base de données. Il est donc important de bien s'assurer de la modification que l'on veut effectuer avant de le faire. Il n'est pas besoin de sauvegarder les changements, ces changements dans une base de données sont effectués immédiatement et ne demande aucune opération ultérieure.

FAQ

Q: Où puis-je trouver une documentation plus à jour (relative à une version plus récente de *Tulip*)?

R: Vérifiez toujours ici https://codigolivres.org.br/docman/?group_id=739 ou sur le site de *Tulip*

(<http://tulip.solis.coop.br/us/index.php>) s'il n'existe pas un manuel plus récent. La version la plus récente a de fortes chances d'être disponible en anglais avant les autres.

Q: Certaines fonctions détaillées ici ne fonctionnent pas sur mon système. Comment cela est-il possible?

R: Ce manuel a été constitué à partir de la version de *Tulip* disponible sur le serveur *CVS* le 4 août 2004. Il est possible que la version officielle ne soit pas totalement fonctionnelle. Si vous aimez chipoter, aller chercher la dernière version ici:

https://codigolivres.org.br/cvs/?group_id=739. Sinon, patientez un peu pour la prochaine version officielle de *Tulip*.