

## OpenDBViewer 1.0.1, lecteur de base de données léger SQLite & MySQL

Posté par [Eric Beuque \(page perso\)](#) le 12/09/17 à 19:05. Édité par 5 contributeurs. Modéré par [Xavier Claude](#). [Licence CC by-sa](#).  
Tags : [sqlite](#), [bdd](#), [mysql](#)

OpenDBViewer est un petit logiciel qui permet dans une interface très simple de se connecter à une base de données SQLite ou MySQL.

Je l'avais d'abord conçu en Python pour mes besoins personnels et professionnels. Ce logiciel a été ensuite converti en C++ avec Qt. Bien qu'il soit opérationnel, beaucoup de choses pourraient l'améliorer. Pour cela, les contributions des utilisateurs et des développeurs sont chaleureusement espérées.

[GitHub](#) (777 clics)

OpenDBViewer utilise un système d'onglets pour visualiser plusieurs tables et bases en même temps.



Après avoir eu la volonté de migrer ce logiciel en C++ avec Qt, j'ai eu l'opportunité de proposer ce travail comme mini projet de stage au sein de mon entreprise, sous ma tutelle. Une stagiaire a donc effectué le gros du travail et après un peu de remise en forme du code, nous avons choisi de mettre les sources à disposition de la communauté sous la licence GNU GPL v3. La compilation nécessite Qt5 base devel' et Qt5 qtranslations. Les chaînes de compilation de mon entreprise me permettent ainsi de facilement empaqueter l'application pour GNU/Linux (Debian), Windows et macOS.

J'avoue que nous avons peu de temps pour travailler sur ce projet, mais nous allons continuer à lui apporter quelques améliorations petit à petit en fonction nos besoins. Vos êtes invités à vous joindre à nous : toutes les contributions et les retours de bogues seront traités dans les meilleurs délais. :)

### Objectif de la réécriture

Posté par [realitix \(page perso\)](#) le 12/09/17 à 21:19. Évalué à 4.

Bonjour Eric,

Merci beaucoup de partager ce logiciel que je vais tester. Un élément a attiré ma curiosité. Pourquoi l'avoir réécrit en C++ ? Est-ce que l'utilisation de Python a entraîné des problèmes. Merci d'avance pour la réponse.

### Re: Objectif de la réécriture

Posté par [Eric Beuque \(page perso\)](#) le 13/09/17 à 00:15. Évalué à 4.

Bonjour, j'ai plusieurs raison à cela :  
- de part mon travail et avec les années, je développe beaucoup sous C++ et Qt, et je suis du coup beaucoup plus à l'aise. Je maîtrise au final très peu Python.  
- j'ai aussi rencontré des problèmes de performances pour la gestion des tables avec grande quantité de données (+100000 enregistrement). Problème que j'avais attribué au wrapper Qt du QTreeView et du coup, je me suis que C++ serait plus avantageux.  
- comme je l'ai dit mon entreprise me permet d'avoir une chaîne de compilation avec CMake et C++ et de générer un build à chaque commit.

Voilà le pourquoi du comment :) )

### Re: Objectif de la réécriture

Posté par [bayo](#) le 17/09/17 à 12:20. Évalué à 1.

Je rajoute que créer ses propres modèles pour QTreeView, ou en faire de la composition, à partir de n'importe quel varappeurs Python (PySide, PyQt4, PyQt5) est assez problématique. Je n'ai pas spécialement creusé, mais le fait que QModelIndex stocke des pointeurs et n'ai pas de destructeur surchargeable semble être le problème de leur côté.

Le fait que PySide2 soit maintenant géré en interne par Qt Company va peut être améliorer cela.

Bref, pour une application exigeante, je ne ferais pas bien confiance aux wrapper Qt.

### Re: Objectif de la réécriture

Posté par [Eric Beuque \(page perso\)](#) le 13/09/17 à 00:28. Évalué à 5.

Et j'ajouterais aussi que la dépendance à Python était problématique en terme d'installation sous Windows. Un installateur tel que nous l'avons fait actuellement facilite grandement l'installation, avec les DLL et les programmes qui va bien.

### DBeaver Community Edition

Posté par [NumOpen \(page perso\)](#) le 12/09/17 à 22:02. Évalué à 4.

Ce serait intéressant d'échanger des idées avec <http://dbeaver.kiss.org>

### Re: DBeaver Community Edition

Posté par [cypuser](#) le 12/09/17 à 23:11. Évalué à 0.

Sauf que DBeaver est en java

### Re: DBeaver Community Edition

Posté par [Eric Beuque \(page perso\)](#) le 13/09/17 à 00:25. Évalué à 3.

Bonjour,

en effet je ne connaissais pas ce logiciel, il a l'air très avancé et semble contenir beaucoup plus de fonction que OpenDBViewer.

Après, notre objectif à nous, est vraiment d'avoir les fonctions vitales et une facilité d'installation pour pouvoir déboguer facilement même en production.

### Bah... Ça marche pô...

Posté par [Yth](#) le 13/09/17 à 09:18. Évalué à 6.

Tout allait bien, téléchargement, compilation, lancement, jusqu'à la connexion MySQL.

En utilisant des paramètres qui fonctionnent très bien en ligne de commande, il me dit :

« Veuillez entrer les informations nécessaires »

Et hop, aucune connexion, aucune information, rien...

Ya rien dans le terminal, et je n'ai pas vu de logs...

Je suis tout déçu !

- o Yth.

### Re: Bah... Ça marche pô...

Posté par [Eric Beuque \(page perso\)](#) le 13/09/17 à 09:44. Évalué à 2.

Zut désolé :D, un bug s'était glissé dans les dernières modifications ! J'ai envoyé la correction dans le dépôt.

### Re: Bah... Ça marche pô...

Posté par [Yth](#) le 13/09/17 à 20:33. Évalué à 2.

C'est bon, ça marche, merci :)

Au premier abord, c'est rapide, puis j'ai ouvert une table avec 25 millions d'entrée, et c'était une mauvaise idée.

Bon, je teste ça !

Yth.

### Re: Bah... Ça marche pô...

Posté par [Eric Beuque \(page perso\)](#) le 14/09/17 à 09:44. Évalué à 1.

Je reconnais que le logiciel n'a pas été conçu pour ouvrir des tables lourdes. C'est pas impossible à terme. Mais il est vrai que nous n'avons pas cette problématique dans nos environnements c'est pourquoi cela n'a pas été un axe de développement.

### Re: Bah... Ça marche pô...

Posté par [sc1s19fr \(page perso\)](#) le 15/09/17 à 12:02. Évalué à 0.

Voir [mon commentaire](#) à propos de Python Blaze

ça marche bien... même avec des gros volumes

### Re: Bah... Ça marche pô...

Posté par [tjpic](#) le 13/09/17 à 12:46. Évalué à 1.

pareil pour moi. Je l'ai téléchargé vers 11h00

### Re: Bah... Ça marche pô...

Posté par [tjpic](#) le 13/09/17 à 14:14. Évalué à 1.

```
0sqlDatabase: OMYSQL driver not loaded
0sqlDatabase: available drivers: 0SQLITE
```

Voilà le message en console

### Re: Bah... Ça marche pô...

Posté par [whity](#) le 13/09/17 à 14:56. Évalué à 2.

Tu dois avoir un paquet supplémentaire à installer (la compatibilité mysql pour qt).

Sous debian, c'est vraisemblablement libqt5sql-mysql qui te manque.

...

Mes commentaires sont en wtfpl. Une licence sur les commentaires, sérieux 7\_6\_0

### Re: Bah... Ça marche pô...

Posté par [tjpic](#) le 13/09/17 à 16:36. Évalué à 1.

et ben...non. Je l'ai, maintenant, mais toujours rien et plus aucun message en console.

### Re: Bah... Ça marche pô...

Posté par [Eric Beuque \(page perso\)](#) le 14/09/17 à 09:32. Évalué à 1.

Bonjour,

peut être la compilation s'est faite avec Qt4 au lieu de Qt5, du coup il manque peut être le paquet libqt4-sql-mysql

Cordialement,

### Re: Bah... Ça marche pô...

Posté par [Trollnad Dump](#) le 13/09/17 à 18:00. Dernière modification le 13/09/17 à 18:11.

Autre point : pour le compilant la page github mentionne qt5 base devel seulement, or il faut également les outils de langues et traductions qt5 qtranslations, qui ne sont pas forcément installés par défaut avec kde plasma, selon les distributions. Il serait peut être utile de le mentionner ?

### Multi-bases

Posté par [Thierry Thomas \(page perso\)](#) le 13/09/17 à 10:25. Évalué à 4.

Je n'ai pas encore eu l'occasion de l'essayer, mais quand tu écris « pour visualiser plusieurs tables et bases en même temps », il y a un truc qui pourrait m'intéresser : j'ai des collègues qui utilisent beaucoup MS-Access, et j'avais tenté de le remplacer par le module équivalent de Libre Office, mais j'ai dû abandonner à cause de ses limitations. Access permet en effet de « lier » des tables de différentes bases MySQL (ou n'importe quel autre SGBD via ODBC), et de construire des requêtes de jointures entre ces différentes tables, même si elles se trouvent dans des bases distinctes. Est-ce que OpenDBViewer permettrait ça ?

### Re: Multi-bases

Posté par [Eric Beuque \(page perso\)](#) le 14/09/17 à 09:25. Évalué à 2.

La version Python que j'avais fait avait un connecteur ODBC, en revanche la version C++ ne l'a pas encore, mais c'est prévu d'être rajouté bientôt.

Après il n'est pas possible directement de faire des jointures entre les bases ouvertes dans le logiciel (sauf avec SQLite qui le permet nativement dans ses requêtes SQL).

### HeidiSQL

Posté par [fwlcat](#) le 13/09/17 à 22:04. Évalué à 1.

Salut,

merci de ce partage.

Perso j'utilise de temps à autre <https://www.heidisql.com>, c'est GPL mais il n'y a que des versions Windows (cependant ça fonctionne plutôt bien sous Wine).

### réécriture

Posté par [PsychoFox](#) le 14/09/17 à 14:31. Évalué à 1.

Question bête. Pourquoi la réécriture en C++ ?

### Re: réécriture

Posté par [Xavier Claude \(page perso\)](#) le 14/09/17 à 14:56. Évalué à 4.

C'est un peu le premier commentaire <https://linuxfr.org/news/opencvviewer-1-0-1-lecteur-de-base-de-donnee-leger-sqlite-mysql#comment-1713150> :

...

« Rappellez-vous toujours que si la Gestapo avait les moyens de vous faire parler, les policiers ont, eux, les moyens de vous faire taire. » - Coluche

### Python

Posté par [sc1s19fr \(page perso\)](#) le 15/09/17 à 11:55. Évalué à 2.

Bonjour,

Pour info voici 2 logiciels en Python avec des objectifs assez similaires à ce projet

En mode console avec curses

<https://github.com/TabViewer/tabview>

En mode graphique avec PyQt

<https://github.com/TabViewer/gtabview>

ça utilise (pour gtabview) Blaze <http://blaze.pydata.org/> et donc ça peut cibler aussi PostgreSQL (et tout ce que Blaze / Odo

<http://odo.readthedocs.io/en/latest/> accepte en entrée )

Bonne journée

Note : les commentaires appartiennent à ceux qui les ont postés. Nous n'en sommes pas responsables.