



## Introduction

[Introduction](#)

[Install BMRT](#)

[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)

A travers cet article, je vais vous expliquer comment d'abord installer les **Blue Moon Rendering tools** et ensuite le logiciel de modélisation **The Mops** sur votre machine Linux. Les **Blue Moon Rendering tools** ne forment pas un logiciel de 3D complet, il ne s'agit en fait que d'un moteur de rendu qui utilise l'interface **Renderman** de **Pixar** pour communiquer. Les **BMRT** sont développés par **Larry Gritz**, employé par **Pixar**, célèbre société d'imagerie qui a réalisé **Toy Story**. La qualité de rendu des BMRT est extraordinaire avec des rendus en radiosité, aujourd'hui Larry peut être fier que son programme soit employé pour réaliser certains courts et longs métrages. Comme les **BMRT** ne sont qu'un moteur de rendu, il nous faut un autre logiciel pour modéliser les scènes. Bien qu'il n'implémente pas toutes les possibilités des **BMRT**, **The mops** est un outil qui peut déjà réaliser quelques petites images. Il existe d'autres modeleurs sous Linux, comme **AC3D** mais ce dernier est assez limité. Il est aussi possible d'utiliser des moulinettes pour transformer des fichiers **.3ds** ou **.max** en fichier **.rib** qui pourront être rendus par **BMRT**. C'est ce que nous verrons à la fin de cet article.

Avant tout, il vous faudra récupérer l'archive sur le site officiel de **BMRT** à l'adresse suivante : <http://www.bmrt.org>. Vous avez plusieurs versions binaires disponibles selon la plate-forme utilisée, en revanche les utilisateurs Linux sur **PowerPC** ne pourront utiliser les **BMRT**. Il vous faudra aussi télécharger l'archive de **the Mops** que vous trouverez à cette adresse : <http://www.informatik.uni-rostock.de/~rschultz/mops/>



## Installation des BMRT

[Introduction](#)

[Install BMRT](#)

[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)

Une fois l'archive récupérée, lancer, dans une console, la commande `tar -xzvf //l'accès_au_répertoire_contenant_l'archive/BMRT2.5h.linux-glibc2.tar.gz` cela décompactera l'archive automatiquement. Vous pouvez utiliser aussi un logiciel avec une interface graphique comme `gxtar` pour décompacter l'archive si la ligne de commande vous fait peur.

Après le désarchivage un répertoire nommé `BMRTx.x` sera créé, il contient un ensemble de fichiers et de répertoires que nous allons répartir maintenant sur votre système.

Passez sous le compte `root`, ou ouvrez une console et faites un `su` pour devenir `root`. Lancez votre gestionnaire de fichiers préféré et vous devez :

- Copiez le contenu du répertoire `/bin` dans votre répertoire `/usr/local/bin`, prenez garde à ce que vous ayez le droit d'exécution sur ce répertoire.
- Copiez `libritout.so` (ou `libribout.a`) et `libslcargs.a` dans votre répertoire `/usr/local/lib`, vérifiez que vous possédez bien les droits de lecture sur ces fichiers.
- Copiez les fichiers contenus dans le répertoire `/include` vers votre répertoire `/usr/local/include`, vérifiez que vous possédez bien les droits de lecture sur ces fichiers.
- Créez un répertoire `/shaders` dans `/usr/local` et copiez tous les fichiers portant l'extension `*.sl *.slc` et `*.h` contenus dans le répertoire `/shaders` vers votre répertoire `/usr/local/shaders`. Vous devez aussi copier le sous-répertoire `/examples` dans ce répertoire. Vérifiez que vous possédez bien les droits de lecture sur l'ensemble de ces fichiers.

Pour être certain que le programme de rendu `rendrib` trouve bien les différents shaders (fichiers `*.slc *.sl`) nous allons paramétrer une variable d'environnement. Si vous utilisez `ksh`, vous pouvez ajouter la ligne suivante à votre fichier `/home/votre_répertoire_home/.profile` ou `/etc/profile` :

```
export SHADERS=./usr/local/shaders
```

alternativement, vous pouvez ajouter dans votre fichier `/home//votre_répertoire_home/.cshrc` ou `/etc/cshrc` (chez Suse se fichier se nomme `csh.cshrc`) :

```
setenv SHADERS ./usr/local/shaders
```

Lancez la commande `ldconfig` en tant que `root` pour que le système tienne compte de ces modifications.

Pour fonctionner; le langage `slc` (*Shading Language compiler*) a besoin d'un préprocesseur de compilation : `cpp`. Si vous avez installé les paquetages `gcc`, celui-ci devrait être installé et complètement opérationnel. Saisissez `cpp --help` dans une console pour vérifier sa présence et si il fonctionne. Le cas échéant, installez le **compilateur GNU gcc** livré avec votre distribution.

Vous pouvez copier le fichier `.rendribrc` vers votre répertoire utilisateur si vous souhaitez paramétrer quelques fonctionnalités de `rendrib`, sachez que si vous utilisez **The Mops**, vous n'avez pas besoin de vous soucier de ce point.



## Test des BMRT

[Introduction](#)

[Install BMRT](#)

[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)

Pour vérifier le fonctionnement de **rendrib** nous allons lancer quelques calculs d'images qui sont livrées en exemples. Ouvrez une console et saisissez la commande suivante :

```
rendrib -d /usr/local/shaders/examples/teabox1.rib
```



Normalement tout devrait fonctionner et vous devriez voir apparaître cette image en cours de calcul, en effet l'option `-d` permet d'avoir l'affichage du calcul en cours. Vous pouvez calculer les autres fichier **rib** livrés en exemple avec **BMRT**, comme les incontournables théières.

Il est possible que vous deviez copier le fichier `teapot.rib` dans votre répertoire utilisateur pour que les théières fonctionnent (sans faire de jeux de mots :-)).





## Test des BMRT avec Tkmatman

[Introduction](#)

[Install BMRT](#)

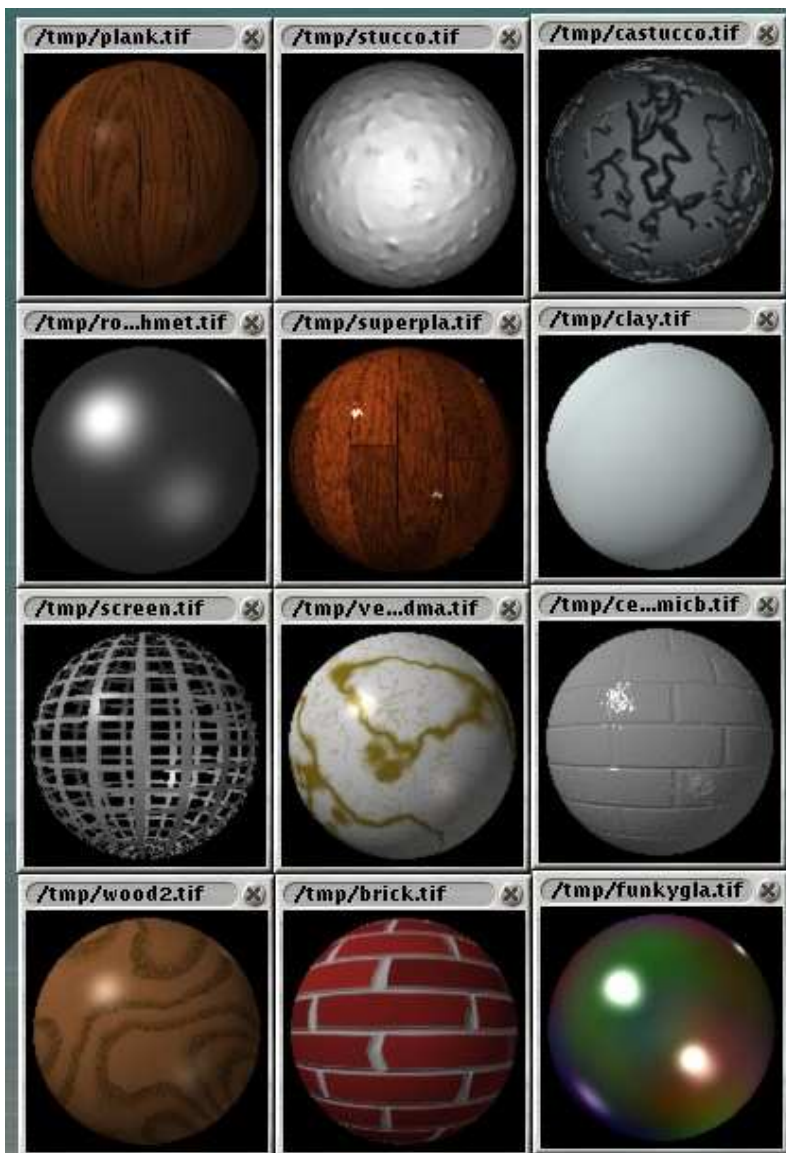
[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)



Vous pouvez aussi visualiser les différents *shaders* livrée avec **BMRT**, les *shaders* peuvent être qualifiés de matériaux extrêmement puissants. Il existe un utilitaire **Tkmatman** que vous trouverez sur <http://www.butz.org/~butz/tkmatman/> il permet de visualiser les différents *shaders* disponibles. Pour l'utiliser c'est très simple, il vous suffit de faire un double clic sur le matériau souhaité et de lancer une prévisualisation avec le bouton test. Attention certains *shaders* peuvent se combiner avec d'autres, ce qui fait que vous pouvez avoir des rendus peu probants. Les *shaders* peuvent aussi être des éclairages, ce qui explique pourquoi vous n'avez parfois rien en prévisualisation.

L'exemple de prévisualisation de matériaux ci-contre a été obtenu avec les matériaux livrés par défaut avec **BMRT**.

Pour l'installation de **Tkmatman**, il suffit de télécharger le script en tcl/tk et de lui donner ensuite les attributs d'exécution pour pouvoir l'exécuter. Vous pouvez directement le copier dans votre répertoire `/usr/local/bin` pour qu'il fonctionne à partir d'une simple console. Tcl/tk est fourni en standard avec toutes les distributions, si ce n'est pas le cas il vous faut installer les paquetages Tcl/tk8 fournis avec votre distribution.



## Installation de The Mops

[Introduction](#)

[Install BMRT](#)

[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)

Les **BMRT** fonctionnent désormais sur votre machine, mais vous ne pouvez rien produire car il n'y a aucune partie modélisation dans **BMRT**. Pour modéliser des scènes qui utilisent l'interface Renderman, vous avez le choix entre Maya, Houdini et Softimage, le seul problème est le tarif exorbitant de ces programmes (environ 100.000 FF soit 15000 Euros la licence par utilisateur). Heureusement un développeur Allemand, *Randolf Schultz*, a lancé le projet **The Mops** qui est une interface de modélisation qui exploite complètement les **BMRT**. Bien que le projet n'est qu'en cours de développement vous pouvez déjà modéliser des objets simples et les obtenir avec le magnifique rendu des BMRT.

Tout d'abord vous devez récupérer l'archive de the mops à cette adresse :  
<http://www.informatik.uni-rostock.de/~rschultz/mops/download.html>

**Randolf** est conscient qu'il n'ait pas aisé pour le débutant de recompiler **The Mops** sur sa machine, alors il en propose une version prête à l'emploi qui ne réclame que les **BMRT** et l'**OpenGL**, c'est une version binaire (binary). Les **BMRT** étant fonctionnels, nous devons juste nous assurer que l'**OpenGL** fonctionne. Pour mieux comprendre l'**OpenGL** sous Linux, il vaut mieux comprendre la situation actuelle. L'**OpenGL** est une bibliothèque qui permet de gérer facilement les objets 3D en temps réel, elle offre un standard de programmation pour tous les développeurs 3D, pour toutes les plates-formes et pour tous les matériels. Ainsi un fabricant de cartes 3D pourra fournir un pilote compatible **OpenGL** qui permettra à la bibliothèque de laisser la prise en charge de certains calculs à la puce graphique : c'est l'accélération matérielle. L'**OpenGL** appartient à la société **SGI** et est donc propriétaire, toutefois un projet visant à réimplémenter exactement les mêmes possibilités existe, il se nomme **Mesa**. En conclusion, soit vous utilisez le pilote **OpenGL** du fabricant de votre carte graphique 3D (c'est le cas des puces **Nvidia**) si il existe, soit votre carte ne prends pas en charge l'**OpenGL**, dans ce cas Mesa s'occupe de faire la même chose mais de façon logicielle (c'est moins performant), soit il existe un pilote **UtaGLX** qui fonctionne avec Mesa et qui prend en charge, en partie, l'accélération matérielle de la puce (c'est le cas de Matrox, **3DFx** et de certaines cartes **ATI**). Dans tous les cas, vous serez plus ou moins contraints d'utiliser le serveur **Xfree4.0x** pour bénéficier des accélérations matérielles. Si le mode logiciel vous suffit, les anciennes versions de X (ex **3.3.6**) sont suffisantes.

Si cette partie configuration de votre carte graphique vous parait trop complexe, assurez-vous d'avoir installé au moins MesaGL avec votre distribution ( vérifiez le avec Kpackage ou GnoRPM ou les logiciels de gestion de paquetages relatifs à votre distribution), dans la majorité des cas cela suffira pour que cela fonctionne.

Après avoir téléchargé l'archive, décompactez-la par un la commande `tar -xzvf mops-0.42d-linux.tar.gz` lancée à partir du répertoire contenant l'archive. Vous devriez avoir un nouveau répertoire portant le nom de **mops-0.4x** . Ce répertoire contient l'ensemble des fichiers nécessaires. A partir de votre gestionnaire de fichier préféré et en tant que root, je vous conseille de copier simplement ce répertoire dans le répertoire /opt ou /usr/local . Il vous suffit ensuite de faire un lien symbolique du fichier /opt/mops-0.4x/src/mops ou /usr/local/mops-0.4x/src/mops vers le répertoire /usr/local/bin pour que mops se lance simplement à partir d'une invite de commandes.



Au premier démarrage de The Mops, celui va chercher les **shaders** de BMRT, comme nous avons paramétré une variable d'environnement pour cela, il devrait vous dire qu'il les a trouvé. Mops contient une zone d'affichage des événements en bas de la fenêtre principale. Vous pouvez donc vérifier que tout fonctionne parfaitement.

Mops cherche aussi le fichier `.mopsrc` qui contient les préférences utilisateurs, mais comme c'est son premier démarrage, il ne le trouvera pas. Mops peut mémoriser la position des fenêtres à l'écran et a besoin de son fichier de configuration pour fonctionner, il vous suffit simplement de faire `file-->save prefs` dans la *fenêtre principale* pour que ce fichier soit créé.

The mops étant fonctionnel, vous pouvez commencer à l'utiliser grâce au [didacticiel](#) disponible pour faire vos premiers pas avec ce programme.



## Conclusion

[Introduction](#)

[Install BMRT](#)

[Test BMRT 1](#)

[Test BMRT2](#)

[The Mops](#)

[Conclusion](#)

[Retour Index](#)

Vous pouvez déjà commencer à faire quelques petits travaux simples avec ces deux programmes, en ayant déjà une belle qualité de rendu qui nécessiterait des logiciels commerciaux onéreux pour obtenir l'équivalent. Il existe quand même d'autres programmes de modélisation sous Linux qui peuvent exporter au format .rib ou qui utilisent directement, à l'instar de Mops, l'interface renderman, vous pouvez trouver :

- **AC3D**: outil de modélisation polygonal simple
- **Sced** : outil de modélisation polygonal assez ancien mais toujours opérationnel.
- **Flow** : Formidable outil qui permet de générer des animations avec des particules. Il comprend un modeleur simple, une interface très soignée et le support des machines SMP, un beau projet, mais un peu complexe à installer.
- **K3D**: encore un projet prometteur mais qui débute seulement, il faudra attendre des versions plus avancées si on veut vraiment commencer à faire quelque chose
- **3ds2rib**: est une moulinette pour transformer les fichiers 3DS en fichiers .rib, vous trouverez également en version d'essais d'une moulinette pour les fichiers .max.

Il existe aussi de nombreuses solutions sous windows, vous trouverez la liste complète sur le site de [BMRT](#).

Dans tous les cas, BMRT est un moteur de rendu qui mérite votre attention car il est relativement puissant, ouvert et bien maintenu que demander de plus ? un bon modeleur sous Linux qui devrait finir par arriver.

N'hésitez pas à nous faire parvenir toutes les images que vous aurez produites avec BMRT, nous les mettrons en ligne avec le plus grand plaisir.

Si vous avez des difficultés pour installer ces deux programmes, n'hésitez pas à me contacter pour m'en faire part.

Amicalement