

Utilisateur de X Window HOWTO

(ed)Ray Briggel/(tr)Adelmar Alain

Table of Contents

<u>L'Utilisateur X Window HOWTO v2-VFrançaise.....</u>	1
<u>Adaptation française du travail de Ray Brigg, ray@aracnet.com.....</u>	1
<u>1. Introduction.....</u>	1
<u>2. Pour Démarrer.....</u>	1
<u>3. Choisir un gestionnaire de Fenêtres (Window Manager).....</u>	1
<u>4. Travailler Sous X.....</u>	1
<u>5. Démarrage de X.....</u>	2
<u>6. Configuration du Window Manager.....</u>	2
<u>7. Polices et Couleurs.....</u>	2
<u>8. Les Ressources X.....</u>	2
<u>9. Clients and Application Tips.....</u>	2
<u>10. Utilisation Avancée de X.....</u>	2
<u>11. Bibliographie et Autres Ressources.....</u>	2
<u>1. Introduction.....</u>	3
<u>1.1 Autres sources d'information.....</u>	3
<u>1.2 Les versions de ce document.....</u>	3
<u>1.3 Réaction et Corrections.....</u>	4
<u>1.4 Remerciements.....</u>	4
<u>1.5 Droits d'Auteur (Copyright).....</u>	4
<u>2. Pour Démarrer.....</u>	6
<u>2.1 Le Système X Window: Historique et Architecture.....</u>	6
<u>2.2 Anatomie de votre Table de Travail (Bureau).....</u>	6
<u>2.3 L'ouverture de X Window.....</u>	7
<u>2.4 Le gestionnaire d'Affichage X.....</u>	8
<u>3. Choisir un gestionnaire de fenêtre.....</u>	9
<u>3.1 FVWM et Ses Ancêtres.....</u>	9
<u>3.2 Le Vaste Monde des Systèmes Window.....</u>	10
<u>3.3 Les interfaces graphiques d'X.....</u>	10
<u>3.4 Les environnements Bureau X.....</u>	11
<u>3.5 Les fringants Gestionnaires de Fenêtres.....</u>	12
<u>4. Travailler dans X.....</u>	14
<u>4.1 Options de Lignes de Commandes.....</u>	14
<u>4.2 Noms d'Affichage.....</u>	15
<u>4.3 XTerm Contre Rxvt, ou, Connaître Son Emulateur Terminal.....</u>	15
<u>5. Démarrage de X.....</u>	16
<u>5.1 Un Exemple de Configuration de Démarrage.....</u>	16
<u>5.2 Un Démarrage Plus Intelligent.....</u>	17
<u>5.3 Obtenir Les Fenêtres Ou Vous Les Voulez.....</u>	18
<u>6. Configuration du Window Manager.....</u>	19
<u>6.1 Configuration de Base de FVWM2.....</u>	19
<u>6.2 Configuration Avancée Dans FVWM2.....</u>	19
<u>6.3 Configuration De Raccourcis Dans FVWM2.....</u>	20
<u>6.4 Thèmes de FVWM2.....</u>	20
<u>7. Couleurs et Fontes (Polices de Caractères).....</u>	21
<u>7.1 Fontes Démystifiées.....</u>	21
<u>7.2 Aliases et Configuration des Polices de Caractères.....</u>	22
<u>7.3 Utilisez les Polices de Type 1 Dans X.....</u>	23

Table of Contents

2. Pour Démarrer

<u>7.4 Utilisation des Polices TrueType Dans X</u>	24
<u>7.5 Couleurs</u>	25
<u>8. Les Ressources De X</u>	26
<u>8.1 X Ressources: Les Bases</u>	26
<u>8.2 À l'Intérieur de la Base de Données de Ressource de X Avec Editres</u>	27
<u>8.3 L'anatomie Des Ressources de X</u>	28
<u>8.4 Rendant Vos Derniers Changements Avec .Xdefaults</u>	29
<u>8.5 Votre Propre Répertoire De Ressource Utilisateur</u>	31
<u>9. Clients et Trucs d'application</u>	32
<u>9.1 Économiseurs d'Écran Pour X</u>	32
<u>9.2 Emacs et XEmacs</u>	33
<u>Gimp</u>	33
<u>10. Utilisation Avancée de X</u>	34
<u>10.1 Bibliothèques et Compilation d'Applications X</u>	34
<u>10.2 Sécurité avec X Les Bases</u>	35
<u>10.3 Plus Au Sujet De l'Autorité De X</u>	36
<u>11. Bibliographie et Autre Ressources</u>	37

L'Utilisateur X Window HOWTO v2-VFrançaise

**Adaptation française du travail de Ray Brigleb,
ray@aracnet.com**

par Alain Adelman aadelmar@free.fr - v2.0, 22 Octobre 1999

Ce document contient des informations sur la configuration de l'environnement de fenêtre de X pour un utilisateur Linux, aussi bien que pour l'administrateur système débutant essayant de faire un choix dans les nombreuses options de configuration et détails de X Window. Une connaissance de base de configuration et d'installation de logiciel est prérequis, de même que la présence de X sur le système d'utilisateurs.

(ndt: Veuillez tenir compte que ce document n'a toujours pas été relu et corrigé "officiellement", donc toutes vos remarques seront les bien venues aadelmar@free.fr.)

1. Introduction

- [1.1 Autres Sources d'Information](#)
- [1.2 Versions de ce Document](#)
- [1.3 Réaction et corrections](#)
- [1.4 Remerciements](#)
- [1.5 Copyright \(Droits d'Auteur\)](#)

2. Pour Démarrer

- [2.1 Le Système X Window: Historique et Architecture](#)
- [2.2 Anatomie de Votre Table de Travail \(Desktop\)](#)
- [2.3 Invocation de X Window](#)
- [2.4 Le Gestionnaire d'Affichage X](#)

3. Choisir un gestionnaire de Fenêtres (Windows Manager)

- [3.1 FVWM et ses Ancêtres](#)
- [3.2 Le Vaste Monde des Windows System](#)
- [3.3 Les Interfaces Graphiques de X](#)
- [3.4 Les Environnements de Bureau sous X](#)
- [3.5 Les fringants Gestionnaires de Fenêtres](#)

4. Travailler Sous X

- [4.1 Options de Ligne de Commandes](#)
- [4.2 Afficher les Noms](#)
- [4.3 XTerm Contre Rxtv. ou, Connaître ton l'Emulateur Terminal](#)

5. DÃ©marrage de X

- [5.1 Un exemple de Configuration de DÃ©marrage](#)
- [5.2 Un DÃ©marrage Plus Intelligent](#)
- [5.3 Disposer les FenÃªtres ou l'on veut quelles soit](#)

6. Configuration du Window Manager

- [6.1 Configuration de Base de FVWM2](#)
- [6.2 Configuration avancÃ©e de FVWM2](#)
- [6.3 Configuration des raccourcis dans FVWM2](#)
- [6.4 Les ThÃªmes de FVWM2](#)

7. Polices et Couleurs

- [7.1 Polices dÃ©mystifiÃ©es](#)
- [7.2 Aliases et Configuration des polices](#)
- [7.3 Utilisation de Polices de Type 1 dans X](#)
- [7.4 Utilisation de Polices TrueType dans X](#)
- [7.5 Couleurs](#)

8. Les Ressources X

- [8.1 Ressources X: Les Bases](#)
- [8.2 L'intÃ©rieur Base de donnÃ©es des Ressources X avec *editres*](#)
- [8.3 L'Anatomie des Ressources X](#)
- [8.4 CrÃ©ation de Vos DerniÃªres Modifications Avec *.Xdefaults*](#)
- [8.5 Votre Propre RÃ©pertoire de Ressources Utilisateur](#)

9. Clients and Application Tips

- [9.1 Sauveur d'Ã©cran pour X](#)
- [9.2 Emacs et XEmacs](#)
- [9.3 Quelques Programmes Utilitaires et Trucs](#)

10. Utilisation AvancÃ©e de X

- [10.1 BibliothÃ©ques et Compilations d'Applications X](#)
- [10.2 SÃ©curitÃ© sous X Les Bases](#)
- [10.3 Plus au sujet des Droit d'accÃ©s \(AutoritÃ©\) sous X](#)

11. Bibliographie et Autres Ressources

- [PrÃ©cÃ©dant](#)
- [Suivant](#)
- [T d MatiÃªres](#)

- [Index](#)
 - [Me Joindre](#)

 - [Précédant](#)
 - [Suivant](#)
 - [T d Matières](#)
 - [Index](#)
 - [Me Joindre](#)
-

1. Introduction

Le système de fenêtres X (*X Window System*) est un système avancé de, travail sur réseau, fenêtrage, environnement graphique, le tout développé à l'Institut des Technologies du Massachusetts (MIT), et réalisé pour la première fois en 1984. Ce document présume que vous avez installé X et ses fonctionnalités. Nous allons apprendre comment utiliser X de façon productive, j'entends par là comme une option lors de l'installation qu'offre les distributions Linux livrées avec X, compilées et prêtes à l'emploi.

1.1 Autres sources d'information

Si vous êtes vraiment débutant, vous pourrez trouver le XFree86 HOWTO et *XFree86 Vidéo Timings HOWTOs* (Synchronisation Vidéo XFree86 HOWTOs) qui peut vous être d'un plus grand secours, vous pourrai le découvrir au même endroit que l'autre. A la fin du document vous trouverez aussi un chapitre Bibliographie et Ressources, pour trouver encore plus d'informations. Oh, et ne pas oublier de lire les **pages du man**.

1.2 Les versions de ce document

Des nouvelles versions de ce document de cet HOWTO peuvent être périodiquement posté au comp.os.linux.help .

Ils seront aussi envoyés vers différents sites Linux FTP & WWW, comprenant le site web: [Linux Documentation Projet](#). La dernière version peut généralement (presque) toujours, être trouvé au: <http://www.croftj.net/~ray/howto> mais le dépôt de Metalab est le point canonique de distribution. (ndt.: nous essaierons de tenir à jours les futures versions qui sont déjà sur la table de travail de Ray Brigleb sur le site fr au: <http://www.linux-france.org/articles/HOWTO/...> ou <http://www.freenix.org/unix/linux/HOWTO/>)

Des changements dans la version 1.1 couvrent un plus grand champs d'application de **rxvt**, élargissent également les champs d'application des Ressources de X, davantage d'historique de X, des références corrigées de SunSite de remarques pour Metalab, et des instructions sur l'utilisation des Polices de Type 1 avec X.

Les changement de la version 1.2 comprennent un champs d'application plus demandé de l'utilisation des polices TrueType dans X, plus de détails au sujet des arrangements de xdm, ainsi qu'une quantité croissante de ressources supplémentaires. Il y a aussi une correction de ma discussion sur les caractéristiques des effacements d'écrans (merci à Heinrich Langos pour la correction!).

Les changements de la version 1.3 comportant quelques stabilisateurs, des infos pour dériver les chemins des polices d'écran dans certaines récentes distributions, et une plus grande quantité d'infos au sujet des bases de la Sécurité sous X. J'ai aussi ajouté quelques trucs et conseils pour l'environnement KDE et une mise à jour

des adresses e-mail

Les changements sur la version (1.4) prennent en compte les corrections et les ajouts de Anthony J., et quelques bon très bons trucs au niveau de la sécurité de la part de Tomasz Motylewski

Les changements de cette version (2,0) inclut des corrections de Guus Bosch, de Brian J. Miller, et de moi-même, aussi bien qu'un bon nombre de nouvelles mises à jour et informations, et une intervention en faveur d'un nouveau défenseur!

1.3 Rétroaction et Corrections

Si vous avez des questions ou des commentaires au sujet de ce document, ne vous gênez pas pour me le faire savoir, par email chez moi au [Ray Brigleb](mailto:Ray.Brigleb@FreeBSD.org) ou à l'adresse "permanente" . Les critiques ou suggestions seront volontairement les bienvenues. Si vous trouvez quelques erreurs ou omissions que ce soit dans ce document, ayez l'obligeance de me le faire savoir de façon à pouvoir les corriger dans la prochaine version. Si il y a des informations que vous auriez aimé voir traiter, ou auquel vous auriez aimé contribuer dans une future révisions, envoyez moi un mail. Je recherche également plus de ressources pour ajouter aux sections et à la bibliographie. Tant que nous y sommes, je recherche également quelqu'un intéressé pour me succéder à l'entretien de ce document, svp faites moi savoir si vous êtes intéressés. Je pense qu'un autre défenseur pourrait s'épanouir dans ce projet que je n'ai eu que tardivement, et juger de la rétroaction que je reçois ceci est toujours un texte très viable et utile.

1.4 Remerciements

Un remerciement spécial au coordinateur du HOWTO, Tim Bynum pour son aide et ses suggestions, et le groupe de XFree86 pour les pages de man et les fichiers d'aides qui accompagnent XFree86. Je tiens réellement à remercier tout de ceux qui m'ont écrit et envoyé leur documentation et leurs descriptions de leurs présents par le passé, j'ai pu à partir de leurs matériels choisir les morceaux les plus pertinents d'une sélection de document. Aussi, un juste remerciement à Matt Chapman, auteur du site du Guide de Window Managers, et à Scott Scriven, pour les trucs et conseils sur le thème de FVWM2 j'ai puisé sur quelqu'un de ces auteurs. Un juste merci rendu à Joe Croft pour ses services Internet, inestimable pour maintenir le développement de ce HOWTO, et à Claire Galper, pour le support moral et ses divers articles et conseils. Le dernier mais non le moindre remerciement à la communauté entière GNU/Linux, pour tout, et Mr. Stallman, pour la rédaction de ce texte (et sa philosophie).

(ndt.: remerciement à Ray Brigleb sans qui j'aurai sûrement couru vers la facilité sans même connaître toutes les possibilités de X et ses gestionnaires qui se suffisent bien sûr, ainsi qu'à Nat Makarevich qui donne toujours les conseils opportuns)

1.5 Droits d'Auteur (Copyright)

Copyright (c) 1998, 1999 by Ray Brigleb. Traduction Copyright 1998-1999

Adapté au français par Alain Adelman en accord avec l'auteur.

[ndt : ce qui suit s'applique à cette traduction également, sûrement, bien que ...?]

Sauf déclaration contraire, les documents HOWTO de Linux sont la propriété de leurs auteurs respectifs. Les documents HOWTO de Linux peuvent être reproduits et redistribués entièrement ou en partie, par n'importe quel moyen physique ou électronique, tant que ses notifications sur ses droits d'auteurs sont maintenues sur toutes les copies. La redistribution commerciale est permise est même encouragée; toutefois, l'auteur aimera

être averti de telles distributions.

Toutes traductions, dérivé de travail, ou incorporation de travaux global ou partiel de n'importe lequel des documents HOWTO Linux doit être couvert au bas de cette notice par ce copyright. Ceci étant, vous ne pouvez pas produire un dérivé de travail à partir d'un HOWTO et imposer des restrictions supplémentaires sur sa distribution. Des exceptions à ces règles peuvent être accordé sous certaines conditions; s'il vous plaît, veuillez bien contactez le coordinateur de HOWTO Linux pour de plus amples informations.

En bref, nous désirons favoriser la diffusion de cette information dans le max. de direction possible. Toutefois, nous désirons que soit maintenue le copyright sur les documents HOWTO, et aimerions bien être avisé de quelques projets que ce soit d'une redistribution du HOWTOs, celui ci en particulier!

Nombre de termes mentionné dans ce document sont des noms commerciaux. Sauf déclaration contraire, toutes les trademarks sont propriétés de leurs propriétaires réceptifs.

- [Précédant](#)
- [Suivant](#)
- [T d Matières](#)
- [Index](#)
- [Me Joindre](#)

- [Précédant](#)
 - [Suivant](#)
 - [Index](#)
 - [Retour sujet](#)
 - [Me Joindre](#)
-

2. Pour Démarrer

2.1 Le Système X Window: Historique et Architecture

Le *X Window System* a été développé dans le Laboratoire de Science Computer à l'*Institut de Technologie du Massachusetts*, en tant qu'élément du projet Athena en coopération avec le DEC, et réalisé la première fois en 1984. Robert Scheifler fût le meneur du projet pour l'essentiel du développement, et les origines de X doivent beaucoup au "W" du paquet de fenêtrage (Windowing package), développé par Paul Asente à *Stanford*. En Septembre 1987, MIT (établissement de technologie dans le Massachusetts) publia la première mise en circulation du X11 que nous connaissons et utilisons aujourd'hui. En même temps que X11R2, le MIT prit commande au *Consortium X*, livré en Janvier 1988.

Plusieurs des idées qui sont entrées dans X Window sont également venues de la recherche du centre de recherches de *Palo*

Alto de Xerox Corporation's (PARC), où elles travaillaient sur des ordinateurs comme le Parc et le Star vers la fin des années 70'. Aucun de ces ordinateurs ne le fit tourner de façon commerciale, mais quand Xerox a démontré les possibilités d'un système de fenêtre sur mesure capable de tourner sur *Smalltalk 80*, les personnes on immédiatement étaient accrochés. Cette série de trois ordinateurs a démontré si bien le WIMP (fenêtres, icônes, menus, pointeur) qu'il engendra une révolution dans le monde informatique instantanément. En l'espace de quelques années beaucoup d'utilisateurs d'ordinateur prirent goût au système de fenêtrage de telle manière, et vous pourriez dire qu'ils n'ont jamais regardé derrière.

X Window est actuellement développé et distribué par le Consortium X, toutefois, une licence libre autorise l'existence d'applications gratuites ou tout au moins peu chères. La version de X utilisé sur Linux est la XFree86. XFree86 est une collection de serveurs pour (système compatible UNIX) UNIX-like OSs sur plates-formes Intel x86. Le travail est dérivé à partir du X386, et il contribue pour une grande part en travail de fond dans la X11R6. Nous pouvons penser à XFree86, à toutes fins pratiques, qu'il soit la fenêtre de X pour Linux, à moins que vous ayez acheté un autre serveur de X.

X Window est bâti sur un grand nombres de trousse à outils, et de bibliothèques (groupe de programmes et instructions servant à l'élaboration de programmes en code source en langage de programmation). Elle est construite sur les qualités intrinsèques et la trousse à outils de développement X (*X Toolkit Intrinsics*) de l'*Athena Widgets*. Bon nombre de programmes utilisent les outils de XView ou ceux de *Motif*. Plus encore font partie toujours d'un système plus nouveau et unifié de fenêtrage et de communication, comme GNOME ou KDE. Si vous trouvez plusieurs de vos programmes ne compilant pas, ou obtenez des erreurs étranges et inexplicables, vous pouvez commencer par vérifier si X à été correctement installé, parce que la plupart de ces bibliothèques (à l'exception de Motif) sont libre et de plus sont pré installées avec X sur votre système

Ce rappeler, elle est appelée *X Window*, pas *X Windows* !

2.2 Anatomie de votre Table de Travail (Bureau)

Il y a quelques principes de base et termes auquel vous devriez vous familiariser par vous même pour vous rendre l'utilisation de X beaucoup plus directe. Ces termes apparaîtront à plusieurs reprises encore dans les pages du *man* (*manuel intégré Linux*) et dans les fichiers d'aides, que l'on suggère que vous consultiez autant de fois que nécessaire.

Le **screen** (l'écran) est votre "desktop entier", et les mots peuvent être employés l'un pour l'autre.

Techniquement il signifie l'affichage visuel primaire que vous regardez avec/sur X, et vous pouvez avoir plus

d'un écran, en fait vous peut avoir plus d'un ordinateur fonctionnant depuis un simple serveur X. C'est au delà de la portée de cet humble document, mais vous devriez vous rendre compte de la distinction, prendre conscience de cette particularité. (Votre écran est votre desktop et votre desktop est votre écran.)

Le **root window** (fenêtre racine) est le fond de l'écran. Il est attribué à l'intérieur d'une fenêtre d'unique nom, il ne se comporte comme aucune autre fenêtre, mais c'est de là que vous lancerez vos application, poserez une image dessus, ou juste une couleur uniforme, sur la fenêtre racine (root window).

Le **window manager** est l'interface principale entre le système de fenêtre X et l'utilisateur. Sans le gestionnaire de fenêtre, le système sera plutôt difficile à utiliser, et ne sera certainement pas un outils très productif. Le gestionnaire de fenêtre pourvoit aux fonctionnalités telles qu'aux bordures de fenêtres, aux menus, aux icônes, dessus de bureau virtuels, aux barres de tâche (buttons bars), barre d'outils (tool bars), et permet à l'utilisateur de l'adapter aux besoins du client à volonté, s'ajoutant souvent à sa fonctionnalité dans le processus.

Le **pointer** est la flèche ou n'importe quel forme donnée qui représente la localisation de la souris (ou autre périphérique de pointage) y correspondant sur l'écran. Le pointeur change souvent pour vous donner par retour de l'information relatif au contexte à quoi vous pouvez vous attendre si vous utilisé la souris à tel point ou à tel autre sur l'écran.

Le **window** est une fenêtre, un cadre, une armature dans laquelle réside n'importe quelle application donnée laquelle est "dirigé" par le gestionnaire de fenêtre. Ceci inclus plein de jolies choses et n'importe quoi excepté la prétendue fenêtre racine (root window). Même des fenêtres qui ne semblent ne pas avoir d'armatures, cadres, des titres, ou n'importe quel type de bordures de la sorte sont contrôlées par votre gestionnaire de fenêtre. la **active window** (fenêtre active) La fenêtre active est la fenêtre que vous employez actuellement, la fenêtre qui recevra le texte quand vous tapez, elle est traditionnellement signifier par le fait que votre curseur souris est pointer sur elle, bien que ça ne soit pas toujours le cas. La fenêtre active est dites avoir le focus, le reste des fenêtres sur votre affichage écran sont "unfocused".

Menus et icônes se comportent dans X de la même manière qu'ils se comportent dans d'autres systèmes de fenêtrage, et ont les mêmes principes généraux qui s'y appliquent. Les fenêtres avec seulement du texte sont appelés **terminal emulators**, xterm pourrai être un exemple, et ceux-ci émulent fondamentalement un affichage de console texte, mais laissez vous séduire par le multiplex et utilisez en plus d'une à la fois, et profitez de plusieurs autres avantages disponibles devant êtres utilisé dans X. Nous débattons d'eux en profondeur plus loin dans ce document.

2.3 L'ouverture de X Window

L'ouverture de X peut être faite de plusieurs manières. Sur votre système il peut être installé pour partir automatiquement, et vous n'aurai pas besoin de lire cette section. La plupart des systèmes Linux, pourtant, actuellement démarre à la ligne de commande sur login, et vous devez déchiffrer ceci par vous même.

La manière la plus fondamentale de commencer X est avec le **xinit**. Ceci vous mènera à un dessus de bureau blanc par défaut (gris en réalité car le gestionnaire de fenêtre n'est pas chargé). Si aucun programme client n'est indiqué sur la ligne de commande, le xinit recherchera le dossier .xinitrc pour fonctionner comme un script de shell (interpréteur de commande), pour le démarrage des programmes client. Si ce fichier n'existe pas, xinit emploiera la commande suivante par défaut:

```
xterm -geometry +1+1 -n login -display :0
```

Utilisateur de X Window HOWTO

Comme vous voyez, ce n'est pas très utile. La manière la plus commune de commencer X est avec la commande **startx**. C'est la plus civilisée des méthodes pour commencer le système de fenêtrage, mais exige que vous ouvriez une session avec un shell en mode textes, et commencez le système de fenêtrage par vous-même. Pour beaucoup d'utilisateurs de Linux c'est la manière la plus commune de commencer X, elle est également la plus flexible. Vous pouvez publier des commandes telles que ce qui suit :

```
startx -- -bpp 8      #start x in 256 color mode
startx -- -bpp 32    #start x in true color mode
```

Les **doubles tirets** passent les arguments directement au **xinit**, et de cette façon vous pouvez lancer le démarrage d' X dans la résolution que votre travail exigera, et néanmoins pouvoir toujours l'employer depuis le fichier de configuration que nous couvrirons plus tard dans ce document.

2.4 Le gestionnaire d'Affichage X

Le programme **xdm** fournit des services similaires pour **getty** et **login**, lequel autorise les utilisateurs à se loguer à l'intérieur d'un système et à démarrer leurs interpréteurs de commandes basic (shell). Néanmoins, si vous démarrez X avec **xdm**, les utilisateurs n'auront seulement besoin que d'entrer leur nom d'utilisateur et leur mot de passe à un sympathique prompt, et ils seront ainsi parachuté directement à l'intérieur d'un environnement graphique. C'est simple et facile à utiliser, c'est aussi fréquemment compris dans les collèges, universités, cyber cafés, environnement de travail, n'importe où vous avez des utilisateurs pas nécessairement familier avec Unix, dans tout les grandes surfaces.

xdm peut être configuré avec des fichiers de configuration localisés dans `/usr/X11R6/lib/X11/xdm` sur votre système Linux. Le fichier `xdm-config` est là pour configurer la manière dont le login de l'écran apparaîtra à l'utilisateur, et **Xsetup_0** est utilisé pour enseigner à **xdm** quels programmes devons être lancés au démarrage de X. Une certaine partie de la configuration d'un utilisateur normal peut être mise dans leur fichier `.xinitrc`, si **xdm** est sollicité pour une utilisation normale.

Un exemple de fichier `Xsetup_0` est placé ici pour voir, lequel peut vous aider pour configurer votre système. Le programme **xfstt** est le serveur de police **TrueType** (nous verrons ça plus loin dans ce document). Aussi, notez que nous utilisons un script shell (interpréter de commande interactif) ici, et il essaie d'appeler **xv** pour établir le fond d'écran pour une belle image (à la place de l'ennuyeux modèle blanc et noir), et si sont appel échoue, **xsetroot** est appelé, pour un dernier essai en vue d'établir le fond d'écran vers une belle couleur bleu (ou autre à votre convenance, nous verrons plus loin le nom des couleurs afin quelle soient reconnues lorsque vous les appelez sous X).

```
#!/bin/sh
xconsole -geometry 480x100-0-0 -daemon -notify -verbose -fn \
'-schumacher-clean-medium-r-**-10-**-***-***-' -exitOnFail
/usr/X11R6/bin/xfstt &
/usr/X11R6/bin/xv -quit -root \
/usr/local/share/WindowMaker/Backgrounds/InDreams.jpg \
|| xsetroot -solid darkblue
xset fp+ unix/:7100
```

Beaucoup de distributions de Linux incluent ce service automatiquement. Si vous êtes capable de voir le niveau d'exécution de votre système Linux, vous pouvez voir probablement que le niveau 3 est le niveau normal de démarrage normal (voir `/etc/rc.d`) à moins que vous n'ayez amorcé en passant par quelque chose comme le **xdm**. Si vous l'êtes, vous allez commencer dans le niveau d'exécution 5. C'est quelque chose de normal sur des systèmes de Linux (et similaire). Les programmes tels que le `linuxconf` et ses équivalents

fonctionnels devraient pouvoir ajuster cela.

Également sur beaucoup des plus récentes distributions Linux les systèmes ont des équipements tel que le gdm et le kdm, qui sont des équivalents avertis pour GNOME et KDE de ce programme. Ceci donc ne change juste que le look et la sensation pour convenir à votre préférences de bureau, les autres versions contiennent plus de dispositifs, tels que se rappeler pour quel environnement de bureau vous aviez opté la dernière fois, les capacités à l'arrêt et la réinitialisation de la console, et ainsi de suite.

-
- [Précédant](#)
 - [Suivant](#)
 - [Index](#)
 - [Retour sujet](#)
 - [Me Joindre](#)

[Suivant](#) | [Précédant](#) | [Retour sujet](#)

3. Choisir un gestionnaire de fenêtre

Ceci est une réflexion sur les gestionnaires de fenêtre disponibles à l'utilisateur de fenêtre X, avec en guise de choix une comparaison affinée. Dans l'ordre de parution sur le marché pour empêcher ce document d'influencer par quelque moyen que ce soit, il se veut impartial (et il n'est pas aisé de le maintenir, ainsi) J'ai censuré les commentaires style " le plus populaire " ou " le meilleurs des gestionnaires de fenêtre ". Je crois que cela sera plus représentatif pour nos objectifs, et une fois que vous aurez compris les concepts de base de quelques uns, vous aurez plus ou moins arrêté votre choix sur eux tous, ou vous serez encore plus embarrassés.

Votre choix sur le gestionnaire de fenêtre peut considérablement influencer ce que sera votre expérience informatique et le plaisir que vous en tirerez. Vous passez beaucoup de temps en face de votre fenêtre lorsque vous êtes sur X, et vous ne voulez pas, dans une certaine mesure et c'est bien naturel, que ce soit par trop rébarbatif. Certains gestionnaires de fenêtres sont personnalisables à volonté, tant qu'à faire d'avoir une nouvelle OS (Operating System) autant être agréablement surpris . Si vous n'aimez pas les icônes, débarrassez vous en ! Si vous manquez de barre d'outils, vous pouvez vous en construire une, elle sera comme vous le souhaitez et ça, ça vaut la peine occasionnée pour la confectionner.

3.1 FVWM et Ses Ancêtres

Les plus communs des gestionnaires de fenêtres de nos jour, tire ses racines dans le gestionnaire de fenêtres FVWM de Robert Nation. Ceux ci incluent FVWM, FVWM2, FVWM95, et beaucoup d'autres. FVWM lui même est dérivé partiellement de TWM, lequel découle de l'installation de X Window, mais que nous ne traiterons pas ici. La syntaxe et l'usage de TWM est similaire à celui de FVWM, mais utilise effectivement beaucoup plus de mémoire sans vraiment être assez souple, ce qui est plutôt gênant pour beaucoup d'utilisateurs.

FVWM (*) est l'original, le vieux standard des gestionnaires de fenêtres, favori de Linux (*l'original*). Actuellement c'est la version 2.0 de FVWM qui est devenue stable et d'usage plus général, cette nouvelle version est préférée, en raison de la syntaxe qui y est plus directe et simplifiée, tout en étant beaucoup plus flexible, comme il se doit. Pourtant , beaucoup de monde continue à se servir de l'original de FVWM sur leurs systèmes, la raison est qu'il a été très longtemps le "*standard*" - le "*favori*", ainsi la plupart des exemples de ce texte sont également applicables à cette vieille version. Et si ledit "*vieux*" travaille convenablement pour vous, il peut ne pas avoir nécessairement besoin d'une mise à jour, hélas il peut casser vos fichiers de configuration à certain titres, en raison de quelques syntaxes qui ont en effet été changés, mais il n'ai pas entièrement

incompatible. Telle est la nature du progrès.

Personne ne semble tout à fait sûr que le F dans FVWM représente (pas même l'auteur, de ce que je puis dire!), mais le VWM semblerait (correctement) indiquer le directeur (*Manager*) de fenêtre (*Window*) virtuel (*Virtual*), et en effet les séries de FVWM sont les Gestionnaires de fenêtre virtuels, et c'est une de leurs forces. Vous pouvez lier des clefs à n'importe quelle fonction, y compris la commutation entre les dessus du bureau virtuels, et le faire facilement et en toute tranquillité.

FVWM2 (*) est un nouveau standard, comprenant de nombreuses améliorations et caractéristiques non établies dans la version antérieure. Différant des nombreux nouveaux gestionnaire de fenêtres, il travail aussi bien sur 8-bit, 256 couleurs d'affichages (comme on l'a vue dans l'exemple "*l'ouverture XWindow*" de la page précédente), que j'ai utilisé pendant un moment, et qui peu encore être amélioré avec des petits icônes et inclinaisons (gradients), pour paraître pour beaucoup semblable à l'interface de Win98 comme on pourrez ou non vouloir. Cette version est aussi beaucoup plus extensible en général, et permet l'utilisation de menus à thèmes et de menus dynamiques. Techniquement, FVWM2 est encore une version bêta, mais il travaille formidablement bien, et je n'ai pas encore eu un seul problème avec lui.

Ressources:

Le site officiel de **FVWM** et de **FVWM2** est <http://www.hpc.uh.edu/fvwm/>. La dernière version peu toujours être disponible au : <ftp://ftp.hpc.uh.edu/pub/fvwm/>.

3.2 Le Vaste Monde des Systèmes Window

Pour les utilisateurs plus familiers avec d'autres environnements de fenêtres sur d'autre systèmes, Il y a une abondance de gestionnaires pour émuler le bureau (*).

Le gestionnaire de fenêtres [icewm](#) est ajusté à une une réalisation et un look plus conséquent, et peut émuler beaucoup d'autres systèmes de fenêtrages standards, touches spéciales de liaisons (hot key bindings), et autres. Les fonctionnalités de OS/2 sont particulièrement agréables. [AmiWM](#) est un type de gestionnaire de fenêtres Amiga Workbench, [mlvwm](#) est un émulateur MacOS. [dfm](#) aussi est une agréable adjonction de ces gestionnaire de fenêtres, un bureau et un gestionnaire de fichiers qui ressemble agréablement à l' interpréteur de commande d'un bureau OS/2 (Workplace Shell.)

Ressources:

Le Web Site des *Window Managers* peut être établi au <http://www.PLiG.org/xwinman/>, est une bonne ressource logique pour trouver tout au sujet des diversités des gestionnaires de fenêtres disponibles, particulièrement pour trouver de bons Gestionnaires de fenêtres pour faire fonctionner d'autres environnements de fenêtrages. De nombreux autres encore sont listés au <http://www.PLiG.org/xwinman/others.html>, fourni par le même auteur que les précédents liens, c'est à vrai dire une formidable et complète liste , *is'nt it* ? Vous pouvez télécharger les plus populaires gestionnaires de fenêtre des versions de Linux sur les archives de [metalab](#).

3.3 Les interfaces graphiques d'X

Open Look et Motif ont vite tentés de standardiser X Window, et le rendent utilisable dans des d'environnements plus traditionnellement répandus, autour d'une importante variété de postes de travail. Tout

les deux sont quelques peu couronnés de succès dans leurs tentatives (dans le look et la réalisation si ce n'est politiquement) et peuvent être utilisés sur des systèmes Linux modernes. Exiger les bibliothèques spécifiques, qui peuvent être employées pour compiler une variété d'applications spécifiquement conçues pour l'un ou l'autre des environnements, pour donner à tous les programmes, un aspect et un touché similaires lesquelles peuvent être utilisés pour compiler une variété d'applications spécialement désignées pour l'un ou l'autre des environnements,

Démarrage avec X11 Release 5, l'interface graphique d'OpenWindows de Sun Microsystem's été disponible pour les utilisateurs d'X. Celle ci inclue deux versions de l'*Open Look Window Manager*, **olwm** et **olvwm** (avec un bureau virtuel). Ça a été développé par Sun dans l'espoir de faire un environnement standard de fenêtrage, et en effet il été un standard avec Solaris pendant plusieurs années. Si votre distribution de Linux ne vous suffit pas, allez y avec le système de fenêtres installé, rappeler vous que vous devez installer la bibliothèque de Xview avant pour travailler, puis vous aurez aussi à placer le répertoire `/usr/openwin/...` dans votre chemin de recherche.

Si vous désirez l'actuelle boite à outils **OSF/Motif Toolkit** pour Linux, vous aurez à payer, le programme et la Boite à Outils ne sont pas Libre de droits. Toutefois, les programmeurs (Hungry Programmers) ont écrit LessTif, lequel permet de compiler des programmes de Motif plus ou moins comme si vous possédiez une boite à outils. LessTif est un **clone** de la boite à outils (*Motif toolkit*). Actuellement **LessTif** est partiellement réalisé avec la plupart des API en fonction. De nombreux programmes déjà travaillent sous cette version libre de la boite à outils, et ils égalent les nouveaux venus avec les gestionnaires de fenêtres, dérivés du code de FVWM, lesquels vous auriez juré auraient été fait depuis le Motif Window Manager.

La caractéristique la plus utile de cette boite à outils, toutefois, consiste en une compilation de programmes dépendant sur votre bibliothèque Motif sur votre système. Le gestionnaire de fenêtre n'a rien de spectaculaire, il n'est principalement utile que dans le cas ou vous émigrez de l'original Motif, il a besoin pour exister de vos fichiers de configuration. Pour toutes intentions et efficacité, vous découvrirez que FVWM est beaucoup plus gorgé de caractéristiques et aussi plus utile, son look et son comportement sont à peu près identique, même reconnaissance de la fenêtre conseil fourni par des programmes construit avec la boite à outils Motif (toolkit).

Ressources:

Les Programmeurs affamés (*Hungry Programmers LessTif*) LessTif peuvent être trouvés à <http://www.lesstif.org/>.

3.4 Les environnements Bureau X

La seconde génération de gestionnaire de fenêtres Linux a été provoquée par KDE, et bientôt jointe par GNOME. Ils y a quelques similitudes saisissantes, et quelques grandes différences, entre ces deux là, et je tenterai de les relater ici.. La plus importante chose à ce rappeler à ce point et que ni l'un ni l'autre n'est sous n'importe quelle forme que ce soit, un produit fini. Tous les deux sont au début d'un long cycle de développement, et pas encore complètement stable et comme tel, ne sont pas adaptés pour la tâche critique qu'est le travail, à l'heure actuelle, *date de ce document (les choses évolues tellement rapidement)*.

L'environnement de Bureau KDE

(Une référence à partir de leur Home page:) ``KDE est un bureau complètement nouveau, incorporant une large suite d'applications pour les station de travail UNIX. Pendant que KDE inclut gestionnaire de fenêtre, gestionnaire de fichier, panneaux, centre de contrôles et beaucoup d'autres composants que les uns espèrent

pour être une partie de environnement de bureau moderne, la réelle force de cet exceptionnel environnement s'allonge dans l'inter-opérabilité de ces composants."

L'environnement KDE Desktop est une tentative de construction d'un environnement de bureau, pas simplement un gestionnaire de fenêtres. Les outils de KDE fonctionnent ensemble tellement bien, pour illustrer, celui ci peut être raillé dans la pensée de KDE être un OS entier. Tout les outils de KDE pour travailler dans un système de fenêtrage sont inclus, et beaucoup plus ont été portés à l'environnement KDE. KDE a déjà atteint un niveau considérable de maturité, mais beaucoup sont réticent pour l'installer sur leur bureau, à cause des conditions d'autorisation de la boîte à outils, sur laquelle KDE est basé. Celui ci a changé un peu dernièrement, et la licence actuellement se qualifie comme Open Source par définition, mais n'est pas la même que celle des logiciels GNU.

Pour ceux qui ne sont intéressés que par le coté travail, business, KDE est souvent la route à suivre. Il y a un certain temps déjà que ce projet à eu lieu, et a su ce faire remarquer pour sa capacité à obtenir des finalités en soit (le fait de pouvoir être auto indépendant, de ce suffire). Dans beaucoup de circonstances vous pouvez convenir de vous moquer en pensant que l'emploi d'un produit d'allure semblable en bonnes parties à Windows. Ce qui peut être une bonne chose. Mais elle peut également susciter quelque frustration, ceux qui ont attendus après quelque chose de sensiblement différent du paradigme sur lequel KDE est fermement basé.

GNOME: L'Environnement GNU (GNU Object Model Environment)

(En référence à leur Home page:) ``GNOME se dresse pour GNU Network Object Model Environnement. Le projet GNOME à l'intention de bâtir un intégré, bureau user-friendly basé entièrement sur des logiciels libres. GNOME est une partie du projet GNU, et GNOME est une partie du mouvement de l'OpenSource(tm). Le Bureau sera constitué de petit utilitaires et de larges applications qui permettrons aux utilisateurs d'avoir une une réalisation et un look d'enfer. GNOME utilise **GTK+** comme la boîte à outils **GUI** pour tout les applications GNOME-compliant."

Le projet GNOME est une tentative en vue de faire plus que le simple travail comme KDE, mais même un petit peu plus que ça, GNOME est moins attaché à un gestionnaire de fenêtres, pour illustrer, et il est inter opérable pas qu'entre les applications, mais entre ordinateurs et plates-formes, comme il utilise *Common Object Resource Broker Architecture (CORBA)*. Aussi, et pour la plus grande majorité, GNOME est basé sur la trousse à outils GTK+, laquelle est libre et de codes source disponibles, il en va différemment pour les dessous de la boîte à outils de KDE, de ce fait groupe les partisans dans la philosophie même de Linux.

Ressources:

Le site officiel de KDE est au <http://www.kde.org/>. Le site officiel de GNOME est au : <http://www.gnome.org/>.

Plus détaillé encore les informations concernant les thèmes ce rapportant à GNOME peuvent être trouvés au : <http://www.gnome.org/gnomefaq/FAQ.txt>. Le Site de CORBA est ciblé au : <http://www.corba.org/> et celui du toolkit de GTK+ est au : <http://www.gtk.org/> .

3.5 Les fringuants Gestionnaires de Fenêtres

Utilisateur de X Window HOWTO

La dernière génération de gestionnaire de fenêtre est très, très jolie, en effet. Mettant en valeur chaque convenance et émulant les logiciels d'exploitation les plus beaux jamais utilisés sur les postes de travail les plus magnifiques dans le monde, ce sont des gestionnaires de fenêtre à lancer si vous disposez d'assez de mémoires et de cycles d'unité centrale à brûler.

Window Maker

(Une citation du home page:) ``*Window Maker est le gestionnaire de fenêtre de X11 conçu pour donner le soutien additionnel d'intégration pour les applications de GNUstep. Il tente d'émuler la réalisation et le look élégant de l'interface graphique utilisateur GUI NEXTSTEP(tm) . Il est relativement rapides, caractère riche, et facile à configurer et à utiliser.* ''

La grande force de ce gestionnaire de fenêtre est qu'il supporte le bureau GNU, , dans le sens qu'il fait un grand et très joli front end pour GNOME. C'est aussi un des plus facile gestionnaire de fenêtres à configurer, et peut configuré depuis une interface graphique, et supporte le glisser (*drag*) d'OffiX et le protocole déposer (*drop*), le changement facile de thèmes du bureau, et il est maintenant disponible à l'intérieur de la populaire distribution Red Hat, aussi il est commode de commuter à partir de FVWM quand l'humeur s'en fait sentir. Comme pour la version 0.50, Window Maker se conforme aux exigences de KDE, aussi bien.

AfterStep

(Une référence du home page:) ``*AfterStep est un Window Manager (gestionnaire de fenêtres) pour X lequel est démarré par émulation émanant du look et réalisme de NEXTSTEP, mais lequel a été significativement modifié selon les requêtes des nombreux utilisateurs. Beaucoup d'adeptes vous dirons que NEXTSTEP n'est pas seulement l'interface la plus plaisant visuellement, mais aussi une des plus fonctionnelle et intuitive. L'idéal de AfterStep est d'incorporer les avantages de l'interface NEXTSTEP, et y ajouter d'utiles fonctionnalités supplémentaires. Les programmeurs d'AfterStep ont aussi travaillé très durs pour assurer la stabilité et un petit programme empreinte (footprint)..*''

Enlightenment

Enlightenment est plus qu'un simple gestionnaire de fenêtres, il est un extrême, dans le détail, et dans la configuration de son environnement, il est également particulièrement attractif en ce qu'il permet aux produits de deuxième zone et des formes de fenêtre complètement personnalisables. Il est accessible en dessin, et au lieu de dicter une police il permet aux utilisateurs de définir leurs propres polices se penchant sur des détails minutieux et infinitésimal; depuis sa réelle fonctionnalité jusqu'aux travers de ses apparences.

Si vous employez GNOME, vous constaterez que Enlightenment est le gestionnaire de fenêtres par défaut, et en fait, il doit être installé pour que GNOME fonctionne. C'est également fondamentalement de fait la réalisation du dispositifs de GNOME pour l'intégration, faisant de lui le choix le plus pratique pour un desktop (bureau) dans cette situation. Beaucoup d'autres gestionnaires de fenêtre travailleront bien avec GNOME, mais vous constaterez qu'Enlightenment excelle. Malheureusement, il est toujours à l'étude, et fonctionne lentement et imparfaitement de temps en temps. (* plus maintenant)

Ressources:

L'officiel Web Site de Window Maker est au: <http://www.windowmaker.org/>.

L'officiel Web Site d'AfterStep est au: <http://www.afterstep.org/>.

Enlightenment peut être trouvé, d'une certaine mesure ou l'on pouvez le prévoir, au:
<http://www.enlightenment.org/>.

[Suivant](#) | [Précédant](#) | [Retour sujet](#) [Suivant](#) | [Précédant](#) | [Retour sujet](#)

4. Travailler dans X

Dans cette partie nous nous familiariserons avec la démarche clients dans X Window et les procédures de bases impliqués dans l'utilisation du système. X n'est pas une interface intuitive sur, et hors de son propre gestionnaire de fenêtres, il est plus facile à utiliser comme un affichage pour déclencher des programmes depuis une ligne de commande. Un des usages les plus commun de X est juste d'avoir divers fenêtres xterm ouvertes en même temps. Pas exactement agrandi hors des capacités graphiques de l'ordinateur, mais c'est une caractéristique sympa de pouvoir démarrer avec. (*)

La flexibilité et l'utilité de la ligne de commande est si importante, en fait, que vous n'avez pas un réel besoin de gestionnaire de fenêtre. Il est agréable, et vous le voudrez rapide, mais le fait est que vous pouvez tout obtenir même sans lui, avec simplement une ligne de commande et/ou une souris. Si vous l'esquintez, vous serai surpris en comparant l'exacte différence entre les clients et le gestionnaire de fenêtres. Si vous voulez jouer avec, *gardez à l'esprit* que la combinaison de touche **Ctrl-Alt-Backspace** vous permettra de sortir de X Window en un instant.

4.1 Options de Lignes de Commandes

La plupart des programmes X tentent d'utiliser les même noms de base pour les options de ligne de commandes. Toutes les applications écrites en utilisant la boite à outils (MIT X Toolkit Intrinsics) du MIT acceptent automatiquement les options suivantes:

Options acceptées par les applications écrites avec MIT X Toolkit Intrinsics

Options acceptées	trad. fr	signification
-display <i>display</i>	affichage	Cette option spécifie quel server X utiliser. Voir chapitre sur Affichage des Noms pour plus de détail.
-geometry <i>geometry</i>	géométrie	La taille initiale et l'emplacement de la fenêtre, dans un format tel que <i>widthxheight+hoffset+voffset</i> or <i>+hoffset-voffset</i> . Noter que si vous imposez un offset négatif horizontal ou vertical, la fenêtre sera placé et décompté en arrière à partir de la droite ou du bas de l'écran, d'autant.
-font <i>font</i>	police	La police utilisée pour afficher le texte dans votre fenêtre.
-bg color	arrière plan	La couleur utilisé pour l'arrière plan de la fenêtre.
-fg color	façade	La couleur utilisé pour le premier plan de la fenêtre.
-name <i>resource-name</i>	nom de la ressource	Utile pour spécifier le nom sous lequel les ressources pour cette application seront établies. C'est utile pour discerner entre les invocations d'une simple application, par exemple, deux xterms peuvent être nommés différemment tant qu'elles peuvent hériter de différentes ressources basées sur ces noms dans la base de données des ressources.
-title <i>string</i>		

Utilisateur de X Window HOWTO

	titre (en tant que chaîne de caractères)	C'est le <i>titre</i> ou <i>capiton</i> pour être utilisé par la fenêtre sur votre affichage, formé avec une chaîne de caractères, généralement utilisé par le gestionnaire de fenêtre pour mettre un titre signalétique dans le haut de la fenêtre. Ne pas confondre avec le -name option
-iconic	en forme d'icône	Invoquer la fenêtre comme un icône.
-xrm <i>resource-string</i>	ID de ressource (string)	Cette option spécifie un nom de ressource et une valeur pour dépasser n'importe quel défaut qui pourrai déjà être établis. Également utile pour des dispositions de ressources X qui n'ont pas eu des options de lignes de commandes explicite. Ex la ligne de commande: <code>xterm -xrm "xterm*background:blue"</code> est fonctionnellement identique que de taper: <code>xterm -bg blue</code> .

4.2 Noms d'Affichage

Chaque Serveur X a un nom affichage de la form (feuille, zone de document) :

hostname:displaynumber.screennumber

L' *hostname* (nom d'hôte) désigne le nom de la machine vers lequel l'affichage est actuellement et physiquement connecté. L' *hostname* peut être omit, alors, dans ce cas, le serveur sera choisi sur la même machine. En fait, si vous êtes le seul à employer X sur votre ordinateur, vous préférerez laisser ceci hors fonction de vos spécifications d'affichage et dans ce cas il prendra le nom de la machine (c'est à dire le nom qui est derrière l'arobas [root@toto /] c'est à dire toto).

Le *displaynumber* (numéro d'affichage) doit probablement être zéro, ceci par contre est nécessaire si le Server X contrôle plus d'une unité clavier et moniteur, c'est à dire, un réseaux de terminaux X.

Le *screennumber* (nombre écran) spécifie lequel des moniteurs dans un dispositif d'écran d'affichage multiple sera utilisé.

Suivant cette spécificité, vous ouvrirez une fenêtre `xterm` sur votre machine locale avec l'option **-display :0.0**.

Vous pouvez voir que nous avons omis le *hostname* (nom d'hôte) à partir de l'option, de cette façon la machine actuelle est supposée n'avoir qu'un seul et même client écran. (N.D.T.: Prendre quand même l'habitude de le définir, ça coûte rien et c'est une bonne habitude `-display:lucarne.0.0`).

Sur systèmes Linux, votre variable `DISPLAY` possède votre nom d'affichage, lequel sur mon système est **:0.0**. C'est généralement réglé par `xterm`, ou un des scripts qui démarre X Window pour vous, bien que vous pouvez l'établir vous-même, comme défini au-dessus, utilisez l'option de ligne de commande **-display** en invoquant votre application.

Si vous avez ouvert une fenêtre `xterm` ou `rxvt`, et ensuite optez pour travaillé en super utilisateur via la commande `su`, vous trouverez que si vous essayez de lancer une application X vous n'aurez pas d'affichage pour le faire. Cela paraît stupide, parce que vous siégé juste en face de votre écran, mais le truc pour l'obtenir est de passer une option à la ligne de commande pour le programme comme ceci `-display:0.0`, et il travaillera bien.

4.3 XTerm Contre Rxvt, ou, Connaître Son Emulateur Terminal

Votre choix de **terminal emulator** peut affecter votre expérience dans X presque autant que votre gestionnaire de fenêtres, aussi une petite discussion de vos meilleures options est mérité ici. Un terminal emulator est un programme comparable à **xterm**, traité au-dessus, lequel vous permet d'émuler la simple console de Linux dans X. Vous compterez fortement sur le programme que vous choisirez, aussi il est rentable de le choisir prudemment pour commencer avec. (*)

Si vous avez une carte vidéo lente, la première chose que vous devrez considérer faire est le changement pour **rxvt**. L'emploi de **xterm** est un bon point de départ, principalement parce qu'il est livré en standard dans toutes les distributions d'X, mais il n'ai pas toujours le meilleur choix. Il contient un tas de code hérités, ce qui explique qu'il soit plutôt assez chargé pour un simple émulateur de terminal. C'est partiellement dû au fait qu'il n'est pas juste un émulateur de terminal de texte ordinaire (plain text), mais aussi émulateur de modes graphiques que vous n'utiliserez jamais. A cause de ça, votre choix peu ce porter sur **rxvt**, et aussi parce que **xterm** peut être extrêmement lent. Je ne suis pas sûr du pourquoi de sa lenteur, mais si vous avez un carte vidéo lente vous comprendrez ce que je tente d'expliquer, et vous serai étonné de la différence quand vous abandonnez **xterm**.

Certains des autres avantages de **rxvt** comprennent des arrières plans pixmap, et un très beau scrollbar (ascenseur). Si vous spécifiez l'option `-pixmap` sur la ligne de commande (en ayant un support compilé sur votre version!) vous pourrai obtenir n'importe quel image `.xpm-type` dans votre fond d'écran. Une caractéristique très cool, et qui étonnamment ne ralenti pas vos sortie pour n'importe quoi, pourtant il restera plus rapidement que votre **xterm** window. Donner lui une chance, je ne l'ai pas essayé depuis un moment mais je pense qu'il a sa place sur Metalab.

[Suivant](#) | [Précédant](#) | [Retour sujet](#) | [Suivant](#) | [Précédant](#) | [Retour sujet](#)

5. Démarrage de X

Nous prouverons, pour les exemples suivants, que nous avons sélectionné quittement un gestionnaire de fenêtres stable, tel que **FVWM2**, pour essayer jusqu'au bouts certains exemples de configurations. je suggérerais de ne donner qu'un exemple de sujet comme objectif de l'étude, car, comme la plupart de ce que nous apprendrons ici et dans les tapes suivantes, l'exemple s'applique à n'importe lequel des Window Manager extérieurs, mais le problème parait plus facilement soulevé en utilisant FVWM2. (*)

5.1 Un Exemple de Configuration de Démarrage

Notre première tape est d'écrire nous même un fichier d'initialisation pour X lui même. Ce fichier peut être soit un fichier au niveau système entier, au quel cas il pourra probablement être placé dans `/var/X11R6/lib/xinit/xinitrc`, ou il peut être imposé sur une base personnelle (par l'utilisateur) par placement du fichier `.xinitrc` dans votre répertoire home. Généralement, il faut s'attendre ce que ce soit par défaut un fichier basic dans la localisation du système attendu (*), peut être renforcé si nécessaire pour des raisons de sécurité, mais mise à part cela les utilisateurs désireront probablement configurer leurs fichiers eux mêmes.

Premièrement permettre dans votre répertoire la création d'un fichier appelé `.xinitrc`. Ouvrez votre éditeur de texte "favori", et coller ce qui suit, ou quelque chose comme ça, dans ce fichier:

```
#!/bin/sh
```

Utilisateur de X Window HOWTO

```
# si vous êtes français ( * ) ne considérer la ligne
suivante qu'au
# point de vue purement théorique est laissé la en
commentaire.
# xmodmap -e "keysym BackSpace=Delete" -e "keysym
Delete=BackSpace"

xsetroot -solid darkslateblue

# démarrer quelques applications basiques
xclock -geometry 96x96+2+2 -bg grey40 -fg black -hl white &
xload -geometry 120x96+2+147 -bg grey40 -fg white -hl darkred
-update 4 &
xterm -sb -ls -geom 80x25-2+2 -title "shell" &
xterm -sb -ls -geom 80x25-2-2 &

# démarrer le window manager
/usr/X11R6/bin/fvwm2
```

Il y a une abondance des choses à apprendre de cet exemple. Tout d'abord, ce dossier sera un script shell, comme indiqué par la première ligne. La commande de **xsetroot** sur la deuxième ligne retourne le fond de notre dessus de bureau avec une couleur bleu plaisante, ce qui n'est pas une mauvaise idée vu que l'on va avoir cette couleur devant les yeux toute la journée.

La troisième et quatrième ligne appellent quelques programmes que j'aime laisser tourner pendant quand je travaille dur. Vous remarquerez que certaines des options faites pour rendre plus agréable le setup, par exemple, en déterminant les couleurs et la géométrie (localisation de l'écran). Je vous donnerai quelques trucs pour évaluer cette tâche plus loin dans un coin. Les cinquième et sixième ligne suivantes également appellent l'ouverture de deux utiles fenêtres **xterm** pour nous, lesquelles nous seront sans aucun doute utiles par la suite.

La dernière ligne est très importante, c'est cette ligne qui démarre votre gestionnaire de fenêtres (window manager!) Noter que les seules commandes qui ne sont pas démarrées en tant que processus d'arrière plan *background processes* (en mettant le signe **&** à la fin) sont **xsetroot**, **xmodmap**, **fvwm2**. Avec ces deux premières c'est sans importance, car le programme sort immédiatement. Mais tout le reste des programmes ont à être activés dans l'arrière plan, sinon quand vous en fermez un, il ce pourrai qu'il tue votre session X Window. Ça ne sera pas très plaisant, ni souhaitable. Comme vu plus haut, lorsque vous fermez **fvwm2**, vous sortez de X.

5.2 Un Démarrage Plus Intelligent

Nous pourrions ajouter plus de fonctionnalités pour notre simple exemple de fichier de mise en route. Pour illustrer le propos, ceci est une bonne méthode pour vous prévenir vous même quand vous pouvez avoir imprudemment démarré X comme utilisateur racine (root). Les utilisateurs de Red Hat semblent faire ceci souvent, pour grand nombre de programmes de configuration lesquels doivent être exécutés en root, doivent aussi être exécutés dans X. Vous pouvez éviter ça par lancement d'une commande **su** pour devenir root durant votre session utilisateur normal dans X, et alors appelant le programme que vous avait besoin d'exécuter comme root avec l'option **-display :0.0** comme débattu plus haut.

```
# change de couleur d'arrière plan pour root
```

5.1 Un Exemple de Configuration de Démarrage

Utilisateur de X Window HOWTO

```
if [ "$USER" = "root" ];
then
    xsetroot -solid DarkRed
else xsetroot -solid DarkSlateBlue
fi
```

Ceci vérifiera si vous êtes bien l'utilisateur nommé root, et si vous l'êtes il vous avertira, en réglant le fond de l'écran sur un rouge dur, plutôt que le bleu amical et usuel. Ce prochain morceau de code est aussi destiné à votre fichier `.xinitrc`, il fusionnera dans vos *ressources utilisateur spécifique* (user-specific resources) et *ressources vaste système* (system-wide resources), première vérification pour être sûr de l'existence des fichiers.

```
userresources=$HOME/.Xresources
usermodmap=$HOME/.Xmodmap
sysresources=/usr/X11R6/lib/X11/xinit/.Xresources
sysmodmap=/usr/X11R6/lib/X11/xinit/.Xmodmap
# Merge in defaults and keymaps
if [ -f $sysresources ]; then
    xrdb -merge $sysresources; fi
if [ -f $sysmodmap ]; then
    xmodmap $sysmodmap; fi
if [ -f $userresources ]; then
    xrdb -merge $userresources; fi
if [ -f $usermodmap ]; then
    xmodmap $usermodmap; fi
```

5.3 Obtenir Les Fenêtres Ou Vous Les Voulez

Tout placage dans votre écran par supposition avec l'option `-geometry` peut devenir en effet très pénible. Particulièrement quand vous considérer que vous avez la possibilité de spécifier l'option `-geomerty` pour positionner n'importe quel programme que vous pouvez exécuter dans X. Ceci permet une grande précision dans le réglage de l'interface, mais, en fait, c'est un hic dans le sens du tas d'options à établir et à choisir.

Ceci apporte une fonctionnalité très agréable aux gestionnaires de fenêtres FVWM. Par défaut, quand vous bouger une fenêtre tout autour de l'écran, vous voyez la spécification de géométrie apparaître sur votre écran dans un petit cadre gris (style bouton). Activez le à l'avance, essayez le déplacement de la fenêtre avec le bouton gauche. Maintenant essayez de la retailler (+ ou - grand). Comme vous pouvez le voir, vous pouvez vous procurer déjà quelques spécifications primitives avec cette méthode. Toutefois, même cette méthode peut être quelque peu difficile, et il sera cool d'avoir tout les détails au sujet de votre fenêtre dans une liste concise.

C'est à ce moment que vous introduirez le programme **xwininfo**. Pour utiliser ce programme, aller dans une fenêtre **xterm** et tapez dans celle là le nom du programme (`xwininfo`). Il vous demandera de cliquer sur une autre fenêtre celle au sujet de quoi vous voulez des informations, et après avoir cliqué dessus elle vous délivrera d'utiles renseignements qu'elle dispose sur cette fenêtre. C'est utile pour la publication d'informations au sujet des fenêtres dont que vous les avez arrangé comme vous les vouliez sur votre écran - exécutez ce programme, puis cliquer sur la fenêtre, dont lors mettez ces paramètres dans votre fichier de démarrage, et votre système de fenêtre sera dorénavant géré dans un état vierge, d'une impeccable précision.

[Suivant](#) | [Précédant](#) | [Retour sujet Suivant](#) | [Précédant](#) | [Retour sujet](#)

6. Configuration du Window Manager

Maintenant nous allons jeter un oeil aux fichiers pour configurer votre Window Manager. Ces derniers varient de gestionnaires en gestionnaires, mais pour notre exemple nous focaliserons sur les fichiers (per-user) tout utilisateurs, et nous allons regarder en direction de FVWM2 dans cet exemple, ainsi vous modifierez le fichier `.fvwm2rc` dans votre répertoire `/home`.

6.1 Configuration de Base de FVWM2

Configurer votre fichier ressources de gestionnaire de fenêtres dans les versions les plus antérieures de FVWM a été plutôt une tâche ardue, parce que les commandes des éléments dans les fichiers nécessitaient d'être très strictes, détaillées et précises, mais cela c'est assoupli en partie dans cette version. A ma connaissance, le seul point crucial du commandement sont les liaisons dynamiques pour les menus, mais nous traiterons de ceci plus loin. Votre installation FVWM2 aura abouti avec un exemple de fichier ressources, et vous devriez être capable de trouver ça dans `/var/X11R6/lib/fvwm2/system.fvwm2rc`. C'est par défaut le fichier de configuration du système étendu (system-wide). Une bonne idée serait de copier ce fichier dans votre répertoire home et de l'appeler `.fvwm2rc`. A partir de ce moment, quand FVWM2 démarre il lira le fichier dans votre répertoire home, et ignorera le fichier de configuration du système étendu.

Maintenant que vous avez obtenu une mise en marche, personnalisé la copie du fichier de ressources de votre gestionnaire de fenêtre dans votre répertoire home, ouvrez le dans votre éditeur de texte préféré, et jetez y un coup d'oeil. Si vous êtes dans X à ce moment, vous pouvez voir la corrélation entre le fichier et ce que vous voyez sur votre bureau. Très utile n'est-ce pas ? Laissons nous tenter par le changement de quelques choses et regardons si nous pouvons le faire paraître un peu plus à notre goût. Probablement la première entrée dans votre fichier (qui n'a pas suivi par le singe dièse # caractère de commentaire) est l'entrée de police de fenêtre (`WindowFont`), suivi d'un très bizarre nom de police. Si vous voulez faire vous faire une idée de comment choisir les variétés de polices et les couleurs dans leur ensemble, sautez à l'avance au prochain chapitre sur les Police et les Couleurs sous X.

6.2 Configuration Avancée Dans FVWM2

Ce que la configuration de FVWM2 peut amener d'incroyablement subtile et complexe. Prenez cet extrait de code à partir du setup de `fvwm2gnome` pour le fichier `.fvwm2rc` :

```
# import config files

Echo -Styles-
Read .fvwm2gnome/config/styles/app.styles
Read .fvwm2gnome/config/styles/window.styles

Echo -Buttons & Keys-
Read .fvwm2gnome/config/buttons.config
```

C'est un exemple d'une méthode pour lire à l'intérieur d'autres configurations. De cette manière vous pouvez moduler votre `.fvwm2rc` -- pas mauvaise idée, considérant comment peu être brouillé un immense fichier de configuration. Ceci peut faciliter le changement et le débogage, et aussi qu'il est plus facile de configurer un si nommé *themes*, sur lequel nous nous attarderons plus tard.

6.3 Configuration De Raccourcis Dans FVWM2

Une méthode réellement pratique pour configurer votre script pour FVWM2 (et FVWM, et **bash**, et un tout un tas d'autre programmes en plus!) est d'utiliser le **Dotfile Generator**, par Jesper K. Pedersen, disponible à partir de <http://www.imada.ou.dk/~blackie/dotfile/>. Vous aurez besoin d'avoir une récente version de **Tcl/Tk** installé (disponible sur la plupart des distributions Linux). Ce programme démarre avec une configuration préétablie, ou prend la première dont vous disposez, et vous permet de la façonner via de nombreuses options de menus structurés.

Soyez prévenu, toutefois, je conseille l'étude de la mise en place de votre configuration par vous même! Non seulement parce que c'est la méthode Unix (pour quoi que ce soit, ça vaut le coup) mais aussi parce que il est plus facile de faire, de petits changements et de manière incrémentale de votre fichier `.fvwm2rc`, de cette manière, et obtenir des choses exactement comment vous les voulez. Aussi, créer un programme **Tcl/Tk**, est un peu plus long. Néanmoins, si vous voulez une méthode moins fatigante pour démarrer avec une configuration décente, Ça vaut nettement le coup.

6.4 Thèmes de FVWM2

Au milieu de nombreuses nouvelles possibilités de FVWM2 se trouve quelque chose communément appelé *thèmes*. Ce qui littéralement signifie que vous pouvez changer le look basic pour tout éléments de vos fenêtre sur votre bureau, en un survol. Noter que si vous avez fait la distinction entre les fonctions de votre gestionnaire de fenêtres et les fonctions des applications elles mêmes, vous réaliserez qu'un thème de gestionnaire de fenêtre n'est pas allé affecter le look et le réalisme des applications elles mêmes. Les boites à outils intégré telle que KDE et GNOME s'occupe d'avoir ces capacités, néanmoins, et les deux peuvent être employées ensemble très efficacement.

Dans l'ordre pour générer un thème, vous devez remonter vos manches et modifier votre fichier `.fvwm2rc` d'un petit morceau de code. Ici se trouve quelque chose que j'ai ajouté à coté du début de mon fichier, essayez de le mettre après les définitions de `Style`:

```
# Blue Theme
DestroyDecor Blue
AddToDecor Blue
+ WindowFont -b&h-lucida-bold-r-*-*-*140-*-*-*-*-*
+ TitleStyle ActiveDown (Solid DarkSteelBlue)\
  ActiveUp (Solid SteelBlue) Inactive (Solid Grey)
+ HighlightColor white blue
+ ButtonStyle 1 -- UseBorderStyle
Style Blue UseDecor Blue, BorderWidth 5, HandleWidth 5,\
  MWMborder, MWMbuttons

# Fonction pour changer toutes les fenetres pour un style nouveau.
DestroyFunc ChangeStyle
AddToFunc ChangeStyle
+ "I" Style $0 $1
+ "I" Recapture
```

C'est une quantité à avaler, je sais. A la base nous sommes à la première place définissant un thème appelé `Blue`, et vous pouvez utiliser ce premier moitié comme un modèle pour désigner d'autres thèmes. Des thèmes peuvent décrire beaucoup plus de caractéristiques que ça, en fait, et peuvent être tout à fait remarquablement différents, depuis un autre. Alors nous sommes entrain de définir une fonction pour changer toute la fenêtre pour un nouveau style. Noter dans les deux sections au-dessus que nous détruisons l'objet

avant de le créer. C'est une bonne idée puisque vous pouvez bien être redémarrant FVWM2 un paquet de fois pour essayer vos différents styles, et ceci le fait travailler un peu plus doucement.

La fonction appelé est nécessaire comme une interface générique pour appeler les définitions des styles que nous avons défini. Maintenant nous créerons les éléments du menu pour les appeler.

```
DestroyMenu "Themes"
AddToMenu "Themes"
+ "Choose a theme..." Title
+ "" Nop
+ "Blue" ChangeStyle "*" "UseStyle Blue"
+ "Mwm" ChangeStyle "*" "UseStyle Mwm"
+ "Flat" ChangeStyle "*" "UseStyle Flat"
```

Ici, nous allons être un petit peu succincts, concis avec la définition de menu, mais il sera beaucoup plus évoqué dans le fichier exemple de votre système. Quoi que nous sommes entrain de faire la fonction ChangeStyle est appelée que nous avons défini au-dessus pour changer le style pour toutes les fenêtres sur l'écran pour un des pré réglage que nous avons défini au-dessus. Remarquez, encore, notre bonne pratique en détruisant le menu avant de le créer. Maintenant si vous redémarrez FVWM2 (vous aurez également un menu option pour ça, si tout va bien!) vous verrez une nouvelle sélection de menus Thèmes, et vous serez capable d'essayer seuls les différents thèmes.

Pour plus d'exemples de Thèmes sur FVWM2, visitez
<http://www.vis.colostate.edu/~scriven/Linux/fvwm/index.html>.

[Suivant](#) | [Précédant](#) | [Retour sujet](#) [Suivant](#) | [Précédant](#) | [Retour sujet](#)

7. Couleurs et Fontes (*Polices de Caractères*)

Il y a de nombreux trucs pour les polices et les couleurs utilisées dans X. Ils ne sont pas tout à fait aussi simples que dans d'autres systèmes, par exemple, la police n'est pas simplement un nom uni terme. Vous indiquez ces ressources tout à fait explicitement, et la tâche semble être plutôt complexe à première abord, mais avec un peu d'expérience vous le ferait en sifflant.

7.1 Fontes Démystifiées

La description logique de fonte dans X (``XLFD``) (*X Logical Font Description*) est le nom complet pour une fonte. Il se compose des champs suivants:

- `foundry` - font foundry..... fondation de la fonte, le particulier ou la compagnie qui a fait la fonte.
- `family` - font family.....famille de la fonte, le surnom général de la fonte.
- `weight` - font weight..... l'importance de la fonte (bold, medium, etc.)
- `slant` - font slant inclinaison de la fonte (italics, oblique, roman (normal), etc.)
- `width` - font width.....la largeur de la fonte (normal, condensed, extended, etc.)
- `adstyl` - additional style.....style supplémentaire (sans serif, serif, etc.)
- `pxlsz` - pixel size.....la taille, le nombre à la verticale de pixels que fait un caractère.
- `ptSz` - similarapproximation de taille (similaire à pxlsz)
- `resx` - horizontal resolution....résolution horizontale en dpi
- `resy` - vertical resolution.....résolution verticale en dpi
- `spc` - space.....espacement, seul utile, à ce qu'il semble, dans les fontes Schumacher

Utilisateur de X Window HOWTO

- `avgWidth` -largeur moyenne des caractères d'un fonte
- `rgstry` - `registry`.....l'enregistrement reconnu qui établit la fonte
- `encdng` -encodage national

De la lumière dans ce chaos, le programme **xfontsel** (le programme de sélection des polices par défaut de X Window) vous paraîtra énormément utile. Essayez de le lancer correctement maintenant. Vous ne trouverez étrangement rien d'utile dans la fenêtre principale, mais essayez de baisser le bouton gauche de la souris sur le bouton `fn dry`. Si toutes vos fontes sont dans l'ordre, vous pourrez voir un menu de sélection tel que `adobe` et `b&h` et `bitstream` et so forth. Sélectionnez en une tel que `b&h` et vous pourrez noter que la fonte, dans la fenêtre réduite, change pour quelques chose d'intelligible. C'est généralement la méthode avec laquelle vous sélectionnez les fontes avec ce programme, démarré depuis la gauche, laquelle est le plus souvent choisi, et déplace vers la droite, pour les options les plus spécifiques. Sélectionnant une option vers la fin la plus à droite ne fera pas sensation avant la mise en fonte, par exemple, est sélectionné, parce que les options sont généralement ordonnées par leurs dépendances sur chaque autres.

Quand vous en serez à choisir à partir de la sélection `fmly`, vous verrez la plupart de options se griser en dehors, et seulement trois demeureront. Cela signifie que ces trois sont la seule famille de fonte faites par cette même fonderie (créateur ou société propriétaire de la fonte). Certaines familles apparaissent sous plus d'une fonderie, par exemple, les deux *Adobe* et *Bitstream* font une variation de la fonte Courier. Maintenant vous pouvez sélectionner le `wght`, et so forth. Après avoir acquis un certain recul vous pourrez adapter la fonte que vous voulez. Vous n'avez pas nécessairement à remplir toutes les options pour choisir une simple police de caractères, il n'y a pas que ces nombreuses polices sur votre système! Les options que vous n'avez pas choisi seront représentées par une `*` indiquant que n'importe quelle option agira dans ce spot.

Quand vous serez content de votre sélection de polices, vous pourrez atteindre le bouton de sélection, et votre sélection sera placé dans le presse papiers X, prêt à être collé dans votre document ou tout ce sur quoi vous travaillé. Par exemple, aller sur votre fenêtre **xterm** et tapez dans quelque chose comme `xterm -font` poursuivant par l'ouverture d'une marque de citation. A ce moment sélectionnez cet endroit sur votre écran, et cliquez sur le bouton central de votre souris (ou cliquez sur les deux le gauche et droit, si vous avez `middle-button impaired`). Cela collera la sélection à partir du presse-papiers, lequel sera la police que vous venez de choisir. Alors entrer le guillemet fermant, et taper `Enter`. Par exemple, un agréable gros **xterm** avec une fonte Courier spécifié pourra paraître comme ça: `xterm -font "-adobe-courier-medium-r-*-14-*-*-*-*-*"` Un **xterm** frais sera le pop up utilisant la police que vous avez choisi.

L'utilitaire **xfd** est très utile pour examiner une police. Si vous le lancer avec une ligne de commande tel que `xfd -fn fixed`, il vous montrera le jeu de caractère pour la police, avec une grande similitude avec les keycaps sur Macintosh. Noter que vous pouvez aussi limiter le nombre de polices que vous voulez, **xfontsel** les affiche avec l'option de ligne de commande `-pattern`, suivi par un référence de spécification de fonte, comme nous l'avons vu au-dessus.

7.2 Aliases et Configuration des Polices de Caractères

Parfois ça devient assommant et vraiment impossible de se rappeler tous les longs noms de fonte. Heureusement, il n'est pas nécessaire de taper cent frappes ou même d'obtenir très exactement le nom de fonte que vous voulez, parce que X fournit quelque chose appelés *font aliases*.

Si vous regardez dans le répertoire `/usr/X11R6/lib/fonts/misc/fonts.alias`, vous trouverez des noms de raccourcis pour de nombreuses fontes. Par exemple, `8x16` est listé comme un raccourcis pour `-sony-fixed-medium-r-normal--16-120-100-100-c-80-iso8859-1`, et n'importe ou vous

Utilisateur de X Window HOWTO

entrez `8x16` en tant que ressource de fonte d'X ou vers une ligne de commande pour un nom de police de caractères, la version longue de la fonte pourra être substitué. Les répertoires *75dpi* et *100dpi* ont des alias similaires pour tirer avantage de, sur la plupart des systèmes la fonte Lucida Sans a un agréable raccourcis.

Si vous ajoutez des polices ou des alias à votre système, vous devrez publier quelques commandes (probablement comme root). Si vous ajoutez des polices, vous devrez publier les deux commandes suivantes (ce sont des exemples, vous devrez mettre dans le bon répertoire, ou peut-être les laisser en dehors, pour faire relire vos polices correctement).

```
mkfontdir /usr/lib/X11/fonts/misc
xset fp rehash
```

Si pour une police vous changez le fichier alias, vous n'aurai seulement qu'à publier la dernière commande ci-dessus, mais ce peut être une bonne idée de les publier toutes les deux, pour être sûr. Avec la commande de `xset` que vous pouvez explicitement publier un chemin de police que vous souhaitez que le serveur utilise, vous pouvez supprimer un répertoire spécifique de votre chemin de police, voir le Home Page pour plus d'information.

Un autre problème commun est que quelques distributions (notamment Red Hat 5.2, pour l'instant) arrivent avec les polices configurées dans un faux ordre. Si vous jetez un coup d'oeil aux votre */etc/XF86Config* (il peut être quelque part ailleurs dans quelques autres distributions, malheureusement, et je ne suis pas sûr où il est sur Red Hat à l'heure actuelle, le localiser avec **locate** peut-être...) Jeter un coup d'oeil à ce dossier si vos polices vous semblent trop laides dans X (très difficile à lire même à de grandes tailles). Vous verrez un groupe d'entrées qui ressemblent à quelque chose comme ceci:

```
FontPath "/usr/X11R6/lib/X11/fonts/misc/"
FontPath "/usr/X11R6/lib/X11/fonts/75dpi/:unscaled"
FontPath "/usr/X11R6/lib/X11/fonts/100dpi/:unscaled"
FontPath "/usr/X11R6/lib/X11/fonts/Type1/"
FontPath "/usr/X11R6/lib/X11/fonts/Speedo/"
```

La chose importante à noter ici est que nous avons démonté les polices bitmapped (polices *graphique établies avec une correspondance de bits*) (les répertoires divers, 75dpi, 100dpi) avant les polices mesurées (telles que Type1 et Speedo). Des polices *bipmapped* établies par bits sont préférées pour le serveur X, parce que les polices mesurées ne paraissent pas très jolies pour un usage commun, elles sont meilleures, peut être, pour des choses comme Gimp ou Netscape. Vous devriez aussi faire un contrôle de bon sens pour voir que tous ces répertoires, en fait existe, et si vous avez modifié récemment, vous pourriez avoir besoin (comme root) entrez et lancez une commande `mkfontdir *` dans le répertoire des **fontes**.

7.3 Utilisez les Polices de Type 1 Dans X

Les types de police qu' X accepte généralement sont d'usage limité, considérant que le seul endroit où vous trouverez des polices de ce type, utilisés, sont dans le système de fenêtres de X lui même, et malheureusement la plupart des drogués de médias et de travail de fontaholics dans les logiciels d'exploitation, préfèrent d'autres formats. **Fontes de Type 1**, le plus généralement utilisées en même temps que des formats de document Post Script, peuvent être trouvées en libre sur l'internet avec une facilité considérable. Essayez

<ftp://ftp.cdrom.com/pub/os2/fonts/> pour commencer.

Se servir de ces polices n'est pas aussi difficile, et graphiques que des programmes comme **GIMP** qui tireront profit d'elles énormément, et le meilleur de tous, les serveurs X de Linux comprennent les polices de Type 1 "out of the box". Pour les employer, déballer d'abord les archives en utilisant les commandes appropriées, et mettez les polices avec une extension .pfb dans `/usr/X11R6/lib/X11/fonts/type1/` sur votre système. Puis, ajouter la référence de ces fontes dans le fichier `fonts.scale` dans cet répertoire, en utilisant le format pour les autres polices, déjà présentes. Puis sauvez le fichier.

Maintenant, vous devriez être capable envers `mkfontdir` de discerner le serveur X au sujet des mises à jour, et puis exécutez

```
xset fp rehash
```

 pour relire le chemin de police. Si ceci ne fonctionne pas pour vous, vous pourriez devoir remettre en marche X pour voir les changements.

Si vous vous trouvez à employer de nombreuses polices de Type 1, et pour des choses autres que X (GhostScript, par exemple, peut manipuler ce format de police aussi bien), vous voudrez vérifier l'utilitaire **typeinst** par James Macnicol. Ce programme peut configurer des polices de Type 1 pour GhostScript et X, et il peut également produire des feuilles témoin pour vos polices, et effectue la majeure partie du travail de la configuration pour vous. Ce programme peut habituellement être trouvé à :

<ftp://metalab.unc.edu/pub/Linux/X11/xutils/>.

7.4 Utilisation des Polices TrueType Dans X

Si vous aviez utilisé un système d'exploitation tel que Windows ou le MacOS, vous pouvez avoir un bon nombre de polices se trouvant déjà sur votre ordinateur ce sont des polices de **TrueType**. Les polices TrueType sont mieux considérées sur les plus petits moniteurs, à basse résolution affichages, tels que votre moniteur d'ordinateur, et tente de fournir d'agréables caractéristiques comme ombrager et anticrénelage, faisant paraître votre affichage plus lisse. Il est également vraiment facile de trouver les polices bon marché de TrueType en vrac, et les CD's contenant plus de 500 polices dans le format de TrueType sont communs dans des boutiques informatiques.

X ne comprend pas des polices de TrueType, et n'a aucune possibilité innée (actuellement) à faire ainsi, et donc il a besoin d'un programme séparé pour faire le *font rendering* (le re-dessin des polices) pour lui. Il y a une bibliothèque de Free Type pour faire ça, mais si vous voulez juste les employer sur votre ordinateur, tout ce dont vous aurez besoin est un programme appelé le **xfstt**, le serveur de police de X pour TrueType. Ce programme devrait être disponible à <ftp://metalab.unc.edu/pub/Linux/X11/fonts/>.

L'installation est assez franche. Extraire les archives, et `make` le fichier, et puis `make install`. Vous pouvez obtenir quelques erreurs qui ne semblent pas prêter à conséquences après la deuxième commande; les ignorer. Ce que vous devez faire après est de créer un répertoire assignable appelé `/usr/ttf` et y mettre certaines de vos polices .ttf là, pour s'assurer juste qu'il va fonctionner. Synchroniser alors le serveur de police avec `xfstt` la commande `xfstt --sync`.

Si vous n'obtenez pas d'erreurs ici, exécutez le serveur de police dans l'arrière plan fond avec la commande `xfstt &` et renseignez votre serveur X11 au sujet du service de police avec la commande `xset fp+ unix/:7100`. Si vous n'obteniez aucune erreur, vous devriez pouvoir exécuter Netscape, le GIMP, même le `xfontsel`, et avoir ces polices disponibles pour vous. Votre plus grand problème peut être de trouver la police que vous voulez, mais ce n'est pas

vraiment une mauvaise chose.

On a porté à ma connaissance que des versions plus récentes de `xset` exigent une version légèrement modifiée de cette commande, en raison des changements du code du `xset`. Si vous lancer une distribution basée sur le Red Hat 6,0 ou postérieure, ou quelque chose d'équivalent, vous voudrez d'abord essayer le `xset fp+ unix/:7101`.

Si tout semble fonctionner très bien, à ce moment vous voudrez sûrement configurer votre système de sorte que les polices fonctionnent quand vous commencez X. Si vous êtes justement en train de démarrer X de la ligne de commande, c'est facile. Tout ce que vous devrez faire c'est d'ajouter les deux commandes du paragraphe précédent à votre fichier `.xinitrc`, dans cet ordre. Lors du prochain démarrage d'X, il devrait travailler finement. Si vous commencez X via `xdm`, vous aurez besoin d'ajouter `/usr/X11R6/bin/xfstt & à /usr/X11R6/lib/X11/xdm/Xsetup_0`. Obtenu cela? Ajouter alors le `xset vp+ unix/:7100` comme il se doit.

Considérer que l'installation d'un serveur de police de TrueType peut être une affaire de ruses. Se rappeler que l'ordre que vous publiez les commandes ci-dessus est important. Le serveur de police doit être commencé avant que X soit commencé, autrement vous courrez au devant des problèmes. Être sûr d'avoir lu le dossier de FAQ qui est arrivé avec la distribution, et le home page de `xfstt` aussi.

7.5 Couleurs

Retournons à notre fenêtre de terminale et essayons encore quelques autres choses. Essayer d'ouvrir un `xterm` avec une ligne de commande comme celle qui suit:

```
xterm -fg darkslateblue -bg red3 &
```

Tant que cette fenêtre n'est pas totalement à votre goût, vous ne pourrez pas donner le meilleur de votre travail dedans, cela fait la démonstration d'un thème intéressant de X, les noms des couleurs. Tant que tout n'est pas très précis, les couleurs sont une agréable méthode pour se rappeler plus facilement qu'une série de nombres hexadécimaux. Noter que les noms de couleur ne sont jamais des case sensibles.

Si vous êtes intéressé par les détails sanglants, ou voulez voir les exemples, ou même vouloir remplacer ces noms idiots de couleur par les vôtres propres pour quelle obscure raison que ce soit, vous pouvez trouver le fichier d'enregistrement qui énumère toutes les couleurs avec leur représentation hexadécimale dans `/usr/X11R6/lib/X11/rgb.txt` sur votre système. Il y a également quelques utilitaires extrêmement secourables avec des noms comme le **`xcolorsel`** et tels. Ils peuvent être trouvés dans les endroits habituels, sur <http://metalab.unc.edu/pub/Linux/> et ailleurs.

Une manière plus précise d'indiquer des couleurs est par la définition numérique. Ceci se compose d'un nom de l'espace de couleur et d'un ensemble de valeurs dans la syntaxe suivante:

```
<colorspace-name>:<value>/.../<value>
```

Un Driver RGB, que vous utiliserez très probablement, est spécifié avec le préfixe " `rgb:` " et a la syntaxe suivante:

`rgb:<red>/<green>/<blue>`, où `<color>` est un nombre d'hexadécimal de 1 à 4 chiffres.

Comme exemple, vous pourriez représenter le rouge de couleur avec `rgb:ffff/0/0`. Pour la compatibilité en arrière, vous pouvez également indiquer le rouge avec une syntaxe telle que `#ff0000` ou `#ffffff00000000`, et vous verrez probablement cela une syntaxe (plus ancienne) beaucoup.

[Suivant](#) | [Précédant](#) | [Retour sujet](#)

- [Précédant](#)
 - [Suivant](#)
 - [Index](#)
 - [Retour sujet](#)
 - [Me Joindre](#)
-

(ndt: Cette page a été traduite par un traducteur logiciel et pas relu donc il doit y avoir des anomalies. (prochainement réécrite))

8. Les Ressources De X

Nous considérerons à ce point quelques modifications plus fines pour votre système. Jusqu'ici vous n'avez obtenu que la configuration du gestionnaire de fenêtres. Le gestionnaire de fenêtres vous laisse adapter tout aux besoins du client qui se produit *en dehors* des fenêtres, les cadres (frames), les bordures (borders), les boutons (buttons), le bureau (desktop)... mais que diriez vous de rendre *l'intérieur* des belles fenêtres? La seule manière de faire ceci est en modifiant les ressources (*Resources*) de X pour votre système.

Les ressources d'X fournissent un mécanisme pour stocker des valeurs par défaut pour des ressources de programme et pour une adaptation en fonction votre environnement de fenêtre de vos aspects et sensations préférés. Les ressources sont spécifiées comme texte de chaîne de caractères qui sont lues à l'intérieur de divers endroits quand une application est exécutée.

Des composants de programme sont appelés de manière hiérarchique, avec chaque objet dans la hiérarchie identifié par une classe aussi bien qu'un nom d'instance. Au top niveau de la hiérarchie ce trouve le nom de la classe et celui de l'instance de l'application elle-même. Par convention, le nom de la classe de l'application est identique au nom du programme, mais avec la première lettre capitalisée (par exemple Bitmap ou Emacs) bien que quelques programmes qui commencent par la lettre 'X' doivent également faire profiter la deuxième lettre pour des raisons logiques (par exemple XTerm ou XConfigurator).

8.1 X Ressources: Les Bases

Essayons quelques exemples pour vous, présenter en douceur au monde des ressources. Commentons par ouvrir une fenêtre xterm. Je suppose que la plupart de vous en a déjà une d'ouverte, sinon vous pouvez en ouvrir une avec un choix de menu, ou bien vous pouvez souhaiter faire un petit retour en arrière dans ce document pour comprendre votre cheminement d'idée! Tapez quelque chose comme ça dans la fenêtre xterm:

```
xterm -background blue &
```

Cette commande devrait avoir eu comme conséquence une autre fenêtre de xterm sautant vers le haut sur votre écran, mais cette fois avec un fond bleu. "Wow," vous dites laconiquement "C'est stupéfiant!" Oui, en effet c'est l'est.

Nous aurons besoin de plus de fond d'écran pour notre prochain exemple. Tapez `exit` dans cet xterm (*qui fermera la fenêtre*) et retourner de nouveau à la fenêtre originale dans laquelle vous avez tapé cette commande.

Essayer ceci: maintenez la touche **Ctrl** et puis maintenez le bouton de souris gauche.

Essayer maintenant cela avec le bouton central de la souris. Maintenant la droite. Toutes les fenêtres de xterm vous donnent des options à l'essai tandis qu'elles fonctionnent. Avec le bouton droits, par exemple, vous pouvez changer la police de la fenêtre. Noter la couleur de ces menus. Essayer maintenant cet exemple:

```
xterm -xrm 'xterm*fontMenu*background: green' &
```

Cette fois, elle apparaît comme une fenêtre normale de xterm. Mais si vous essayez de maintenir Ctrl et le bouton droit de la souris dans cette fenêtre, vous verrez que ce menu (le `fontMenu` mentionné dans l'exemple ci-dessus) est vert. Que s'est-il produit au juste? Si vous regardez les pages du man pour **xterm**, vous verrez beaucoup d'options de ligne de commande, telles que `-background` ou `-font`, que vous pouvez placer quand vous lancez n'importe quelle fenêtre dans X, à côté des options juste pour des fenêtres **xterm**. Ces options particulières sont appelées X Toolkit Options (les options de la boîte à outils de X), et elles s'appliquent pour et au sujet de n'importe quoi dans X.

8.2 L'Intérieur de la Base de Données de Ressource de X Avec Editres

Pour vraiment comprendre l'intérieur des ressources de X, nous devons exécuter **editres**. Allez-y et lancez le (et un programme d'échantillon pour jouer avec) maintenant:

```
xclock &
editres &
```

Vous verrez probablement une horloge simple, et l'apparence plutôt innocente de la fenêtre d' **editres**, blanche la plupart du temps. La fenêtre principale nous montre l'arbre de ressource de n'importe quelle fenêtre donnée sur notre bureau, et par dessus le lancement de programme, il n'y a rien. Ainsi donnons lui en un. Maintenir le bouton gauche sur le menu de commandes, et choisir **Get Tree**. Dans le haut de la fenêtre apparaîtra

Click the mouse pointer on any toolkit client
(cliquer le bouton de la souris sur n'importe quel objet de la trousse à outils).

Ce message est plus que ce qu'il paraît, et c'est un conseil pour nous que quoi pas toutes les applications de X sont des clients de toolkit (cependant la plupart des fondamentales le sont, et le programme nous indiquera habituellement s'il n'est pas dans la page du manuelle). Le curseur de la souris se tournera vers une croix, et vous attend pour cliquer dessus une autre fenêtre.

Pour cet exemple, cliquons d'abord dessus le `xclock`. Vous verrez un couple de choses apparaître dans la fenêtre d'**editres**. Ce sont les branches configurables que la trousse à outils voit. Cliquer sur le bas droit d'une (clock). Elle devrait intervertir. Choisir maintenant "Show Resource Box" à partir des "Commands" menu. Vous verrez une autre fenêtre sauter vers le haut, avec le titre "`xclock.clock.unknown`". Bingo!

D'ici vous pouvez jouer avec les options de configuration de la fenêtre principale de **xclock**. D'abord, cliquer dessus "Set Save File" au fond, et mettre dedans un nom de fichier `/home/votrenom/resources`, pour indiquer que vous examinez certaines des ressources ici. Cliquer maintenant dessus le "foreground" et mettez dans "blue" ensuite pour le prompt "Enter Resource Value:". Frapper alors "Apply" au fond. Les petites entailles des minutes autour de l'horloge devraient tourner au bleu. Aller au "background" et entrez "navajowhite" (ou quelque couleur que vous préférez). Appliquer alors cela. Comme vous pouvez voir, nous pouvons configurer la fenêtre entière juste de cette façon. Mais si vous deviez essayer de lancer un autre **xclock**, elle apparaîtrait juste comme celle-ci apparue avant tous les changements. Nous devons sauver ces changements.

Cliquer "Save" et puis "Popdown Resource Box" (*un nom fantaisiste en effet pour la fonction "Close"!*). Si vous regardez le fichier vous avez juste créé (*cat /home/yourname/resources* peut-être) verrez toutes ces ressources définies, dans le format correct du toolkit de X, pour votre convenance. Mais cela nous fait peu bon, pour si vous essayez de lancer un autre **xclock**, il s'immobilisera sur l'apparence Plain Jane. Aussi voici la dernière et finale étape dans le chargement dans vos ressources:

```
xrdb -merge /home/yourname/resources
```

Cette commande *fusionne* les ressources que nous avons juste créées dans la base de données de ressource de X (**xrdb**) pour votre session de X Window. Cela signifie que pour toutes les futures invocations des applications de X que nous avons modifiées, nos changements auront lieu, et resteront liants. Ainsi si vous lancez un autre **xclock**, il regardera aussi agréable que vous l'avez établi en ce moment, chaque fois que vous le lancez. Faire attention à vous, le **xrdb** est un programme complexe, et vous pouvez vouloir avoir un regard sur les pages du man avant d'y aller, ou jouer encore plus avec lui. Si vous ajoutiez la modification au `.xinitrc` plus tard dans ce document, pour charger les ressources automatiquement sur le démarrage de X, vous aurez seulement à devoir vous inquiéter de la commande de **xrdb** quand vous faites des changements pendant votre session de X.

8.3 L'anatomie Des Ressources de X

Comme vous pouvez voir nous avons traversé une surabondance d'options de configuration ici. Cette méthode de configuration X, comme l'a été conçue, offre des possibilités presque sans limites, et une quantité équivalente de confusion. Pour obtenir une certaine image de l'étendue des ressources dans les X Toolkit Intrinsics seule, lancer les **viewres** de programme, et le jouer un peu autour avec lui. Ce programme montre graphiquement l'arbre des ressources dans la bibliothèque de Xt (*Xt Library*).

Si vous lisez la page du manuel pour X, vous trouverez une définition plutôt obscure de la syntaxe exacte pour définir des ressources qu'elle comprend. Nous pouvons simplifier un bout de ceci tout fait et le décrire dans cette définition essentielle de syntaxe:

```
<program><binding><widget><binding><widget><...><resource>:<value>
```

Cela ne semble pas vraiment si simple, réellement. Bien, définissons quelques choses au sujet de ce qui a juste été dit, et tout commencera à sembler raisonnable après. Vous pouvez jeter un coup d'oeil devant aux exemples dans la prochaine section comme vous lisez le long, si vous voulez.

Le *programme* dans cette définition est l'invocation d'une application dans la base de données de

ressource. Ce serait XTerm pour une invocation de **xterm**, des Emacs pour une invocation de l'éditeur de texte d'emacs, ou un nom défini pour l'utilisateur qui a été donné quand les applications ont été lancées avec le -name d'option de ligne de commande. De cette façon vous pouvez définir les ressources séparées pour les **xterms** lesquelles seront employés dans différentes méthodes. Ce qui est assez cool, vraiment.

L'attache peut être un de deux caractères pour séparer les widgets et tels. Si vous employez un . (point), vous obtenez une attache serrée, qui signifie qu'un widget est directement au-dessus ou au-dessous de l'autre dans la hiérarchie de widget. Ceci a également la priorité la plus élevée des méthodes de spécifications. Si vous employez une * (astérisque), vous obtenez une *prolongue attache lâche*, et vous sauterez tout nombre de widgets dans la hiérarchie, et elle essayera d'assortir le prochain widget possible défini.

Les entrées de widget sont des articles dans l'arbre de widget, par ordre de plus spécifique à moins spécifique, qu'elles apparaissent dans l'arbre de widget, évitent avec des **editres**. N'importe quelle entrée de simple widget peut également être remplacée par un ? (point d'interrogation) pour sauter une définition simple de widget, et assortir tout article possible de widget.

L'article de *ressource* doit être spécifique, et ne peut pas être remplacé par le caractère ? . C'est l'article plus spécifique dans la hiérarchie, et contient habituellement des articles comme la couleur réelle pour définir, police réelle à définir, et ainsi de suite. En fait, tout autrement avant que la ressource dans cette définition puisse être omise et remplacée par un astérisque simple, mais la ressource réelle à définir doit être présente. Si vous mettiez juste un astérisque et le nom plus spécifique de ressource, tel que le `*background: blue`, X essayera de définir cette ressource globalement, pour tous ses clients, si possible.

Après les deux points est l'entrée de valeur. Cette entrée définit ce que la ressource sera placée pour, tel qu'un nom de police ou une valeur de couleur. La valeur peut être indiquée (selon le contexte) comme des données booléennes, numériques, ou des type de données textes (chaînes de caractères). L'entrée de valeur, également, ne peut pas être omise dans une définition valide de ressource.

8.4 Rendant Vos Derniers Changements Avec .Xdefaults

Il y a un fichier magique que vous pouvez mettre dans votre répertoire local appelé Xdefaults. Si vous copiez les lignes dans le fichier ressources à partir du dernier exemple dans les .Xdefaults dans votre répertoire local, vous n'aurez jamais plus à configurer le **xclock** encore! Tandis que ceci ne pourrait pas être l'exemple le plus fin de son utilité, il fait sa remarque. Ce dossier peut être complètement chargé et surchargé de chaque option que vous préférez pour chaque type de programme vous exécutez dans X, et si vous lui prodiguez un soin approprié, vous pourrez toujours facilement y retourner et faire de légères changements quand vous en aurez besoin. Mais la fabrication des choix des changements, et des recherches découlant des ressources subtiles employés par editres peut être un processus extrêmement fatigant et soigneux. En effet, parfois c'est trop de travail, et la plupart de ces ressources vous attendent déjà, au garde-à-vous, groupées par programme, sur votre système.

Utilisateur de X Window HOWTO

Dans le répertoire `/var/X11R6/lib/app-defaults` vous trouverez un grand nombre de fichiers, tous baptisés du nom d'un programme de la trousse à outils de X. Si vous examinez ces fichiers vous constaterez qu'ils contiennent un grand nombre d'options de configuration pour chacun, et je veux dire un grand nombre! Vous ne voudriez pas toutes les options de tous les fichiers dans votre fichier `Xdefaults`, ce serait assomant et tout à fait ennuyeux à traiter. Ce sont les défauts, et il est de ces derniers que vous pouvez décider ce que vous voudriez voir changer pour votre configuration particulière.

Ce qui suit sont quelques échantillons provenant de mon fichier `.Xdefaults`. Noter quelques choses que nous n'avons pas encore mentionnées au sujet des dossiers de définition de ressource. Si une ligne commence par `!` (point d'exclamation), on le considère un commentaire, et le reste de la ligne est ignoré. Si la ligne commence par le nom de fichier de `#include filename`, cette ligne est une directive d'inclusion, et à ce point dans les ressources un autre dossier sera fusionné, quand il est chargé. Ceci peut préserver vos dossiers de ressource d'enfler aussi. Et voici quelques exemples:

```
! Default resources for me@localhost xterms
! start with the generic, move to the specific...
*Dialog*Text*font:      -b&h-lucida-medium-r-*-*12-*-*-*-*-*-*
*dialog*value*background: white
*Dialog*Label*font:     -adobe-helvetica-bold-r-*-*12-*-*-*-*-*-*
*MenuBar*font:         -adobe-helvetica-medium-r-*-*12-*-*-*-*-*-*
*MenuBar*background:   grey80
*MenuBar*foreground:   black
*Label.font:           -adobe-helvetica-medium-r-*-*10-*-*-*-*-*-*
*Label*shadowWidth:    1
*SmeBSB.font:          -adobe-helvetica-bold-r-*-*12-*-*-*-*-*-*
*SimpleMenu*font:      -adobe-helvetica-medium-r-*-*10-*-*-*-*-*-*
*OptionMenu*font:     -adobe-helvetica-medium-r-*-*10-*-*-*-*-*-*
*Command.font:        -linotype-helvetica-bold-r-narrow-*12-*-*-*-*-*-*
*commandBox*font:     -b&h-lucida-bold-r-*-*12-*-*-*-*-*-*
*Toggle.font:         -adobe-helvetica-bold-o-*-*12-*-*-*-*-*-*
*Form.background:     grey70
*TransientShell*Dialog.background: grey70
*Scrollbar.Foreground: grey80
*Scrollbar.Background: grey50
*Scrollbar*cursorName: top_left_arrow
*Scrollbar*width:     16
*shapeStyle:          Rectangle
*XlwMenu.shadowThickness: 1
*shadowWidth:         1

! xterm stuff
xterm*scrollbar.background: grey40
xterm*foreground:     grey90
xterm*background:     grey25
xterm*cursorColor:    white
xterm*visualbell:     on
! rxvt stuff (a quicker, better xterm)
rxvt*color12:          steelblue
rxvt*color15:          white
rxvt*color9:           rgb:ff/7f/5f
rxvt*foreground:       grey90
rxvt*background:       grey10
rxvt*cursorColor:     white
rxvt*font:             lucidasanstypewriter-12
rxvt*loginShell:       false
rxvt*saveLines:        1024
rxvt*title:            shell
```

Utilisateur de X Window HOWTO

```
rxvt*geometry: 80x25
! Make Xman just a little bit more sane
xman*topBox: false
xman*background: lightsteelblue
xman*foreground: black
! xcalc is too bland by default...
xcalc*Command.font: -adobe-helvetica-bold-r-*-10-*-*-*-*-*
xcalc*customization: -color
! Disallow the <blink> tag in Netscape
Netscape*blinkingEnabled: False

! Merge other resources (example)
# include $HOME/.otherXresources
```

Un mot d'avertissement en ce qui concerne des ressources de X, pour les utilisateurs de KDE. Il y a un arrangement dans le panneau de contrôle qui provoquera, la réécriture de toutes vos ressources X, dans vos applications. Ceci peut être un dispositif vraiment agréable, mais ce peut également être un ennui si vous avez soigneusement configuré vos ressources. Ouvrir le panneau de commande et trouver la section "Fonts etc.". Il y a une option au modèle de "Apply du style des application non-KDE APP". Si vous tournez ceci, tous vos programmes adopteront le look et l'ergonomie configuré par KDE (vous pouvez devoir remettre en marche X pour noter ceci). Si vous laissez cette option arrêtée, vos ressources demeureront comme vous les avez configurées.

8.5 Votre Propre Répertoire De Ressource Utilisateur

Vous pouvez également créer un répertoire de fichiers de ressource, exactement comme le répertoire *app-defaults* du système attendu mentionné au-dessus, avec un fichier par programme. Créez le répertoire (pour notre exemple nous utiliserons *app-defaults* sous votre répertoire home) et définissez alors votre variable d'environnement *XAPPLRESDIR* pour se diriger vers elle. Un bon endroit pour placer cette variable serait dans le commencement de votre fichier *.xinitrc*, par exemple, mettre dans la ligne exportation *XAPPLRESDIR=\$HOME/app-defaults* de ligne (si vos fichiers sont pour aller dans un répertoire *app-defaults* sous votre répertoire local).

Maintenant, toutes les fois que vous commencez un programme X, ce répertoire sera recherché par un fichier avec le même nom que le nom de ressource du programme, exactement comme répertoire de système attendu. C'est le nom de client que vous avez utilisé dans des fichiers *Xdefaults*.

Par exemple, un fichier appelé *XTerm* peut contenir la ligne **background: gold*, et tous vos **xterms** pourront, par défaut, se manifester avec un arrière fond couleur or. C'est une alternative agréable pour un simple fichier *.Xdefaults*, et le faire plus clairement lorsque essayant de décider quels arrangements à configurer plus tard, et pour trouver ceux pour un certain programme. Il restent des utilisations pour les *.Xdefaults*, cependant. Il est utile pour placer des ressources non liées à un simple programme, comme les modifications que vous feriez pour tourner tout le certain genre de bleu de bouton, sans se soucier de l'application.

-
- [Précédent](#)
 - [Suivant](#)
 - [Index](#)
 - [Retour sujet](#)
 - [Me Joindre](#)

[Suivant](#) | [Précédant](#) | [Retour sujet](#)

9. Clients et Trucs d'application

Nous avons traité quelques-uns des clients dans X, et ce chapitre en couvrira encore plus. Nous limiterons notre discussion ici aux éléments du noyau les plus fondamentaux et les plus importants, ceux qui viennent avec X ou vous êtes probablement pour chercher vous même leurs utilisations avec X. Si vous avez installé un environnement de bureau intégré tel que KDE ou GNOME, vous en aurez beaucoup d'autres à choisir à partir desquels probablement s'exécutent plusieurs des mêmes fonctions. Cependant, il est important de savoir et de comprendre plusieurs des programmes fondamentaux dans X, parce qu'ils peuvent être très utiles pour travailler avec votre environnement ou du même genre. Aussi X offre beaucoup de nouvelles options dont même vos applications normales de console peuvent tirer profit.

9.1 Économiseurs d'Écran Pour X

Un dispositif commun de beaucoup de logiciels d'exploitation est la capacité de masquer l'écran après une quantité de temps indiquée, et montre sur option certaine sorte d'objet séduisant de demo graphiques, un sauveur d'écran. Il y a des méthodes de couple pour faire cela dans X, aussi.

La manière la plus fondamentale pour employer ce dispositif est en mettant une commande dans votre fichier de démarrage de `.xinitrc` tel que `xset dpms 2400 3600 4800`. Le programme de **xset** peut configurer les dispositifs d'économie d'écran du serveur X, pour ne pas être confondu avec l'écran masquant que le kernel alloue quand vous êtes sur la console de texte. Avec l'option de `dpms`, X peut employer les dispositifs d'économie de puissance de votre moniteur aussi bien. La première option configure combien de secondes sont allouées avant que l'écran ne se masque, la deuxième option est combien seconde avant que le dispositif d'économie de puissance commence, et la troisième option est pour le mode de "OFF". La rotation en une option permet implicitement le dispositif, en plaçant le dispositif à zéro il ce neutralise explicitement.

Beaucoup de distributions de Linux arrivent avec le **xlock** pré installé, ou comme option. C'est un joli économiseur basique et assez agréable sauveur d'écran. Si vous l'exécutez avec l'option `-nolock`, vous pouvez voir certains des modes qu'elle offre, et si vous laissez cette option désactivé, elle vous demandera un mot de passe quand vous déplacez la souris ou appuyez sur une touche, comme dispositif de sécurité. Noter que ce n'est pas une vraie sécurité, parce que à une console de Linux un utilisateur pourrait remettre en marche l'ordinateur ou sortir hors de X avec une combinaison des frappes. La dernière méthode peut être handicapée, cependant, et si vous employez le **xdm**, elle offrira autant sécurité que votre ouverture, ainsi il se peut que seulement le fait de rebooter laisse quelqu'un dedans.

Un plus récent et meilleur programme est **xscreensaver** par Jamie Zawinski. Ce programme offre une grande finesses et précision, exemple, il peut exécuter ses processus à un niveau de propriété plus agréable, diminuant la charge au système pendant qu'il fonctionne, il détecte automatiquement quand l'écran a été mis hors tension par le **xset** et ne fait pas perdre de temps au processeur. En outre, toutes les routines graphiques qu'il appelle sont des demos modulaires, et vous pouvez ajouter des routines sans mettre à jour le paquet entier, et il peut également appeler d'autres programmes, tels que `xearth` ou `xdaliclock`, comme modules.

La dernière version du **xscreensaver** peut être trouvée à <http://www.jwz.org/xscreensaver/>. une fois que vous l'obtiendrez, installez le et préparez le pour y aller, voici quelques ajouts sympas que vous pourriez souhaiter ajouter à votre dossier de Xdefaults:

Utilisateur de X Window HOWTO

```
!!! quelques exemples de XScreenSaver par defauts
! Time out apres 3 minutes, apres mode cycle chaque 2
xscreensaver.timeout: 3
xscreensaver.cycle: 2
! Executer en tres basse priorite, et eteint entre modes
xscreensaver.nice: 12
xscreensaver.fadeSeconds: 2
```

xscreensaver est aussi agréable à avoir sous la main car il est utilisé par le bureau de Gnome desktop comme sauveur d'écran par défaut, et il s'intègre au panneau de contrôle de Gnome.

9.2 Emacs et XEmacs

Si vous êtes fan de l'éditeur de texte **EMACS**, ou juste quelqu'un qui l'utilise souvent, vous trouverez votre travail simplifier dans X Windows. Si vous n'avez pas essayé XEmacs, vous pouvez vouloir l'obtenir pour une utilisation dans X. Il y a des fonctionnalités dans XEmacs qui sont agréables même si vous n'êtes pas dans X, par exemple, votre texte peut être coloré pour assortir le modèle d'améliorations que vous éditez automatiquement. Vous devriez donner la modification suivante à votre fichier `.emacs`, et lire les pages info pour plus d'options. Rechercher également une option pour éditer des visages dans les menus.

```
(global-font-lock-mode t)
(setq font-lock-maximum-decoration t)
```

A noter que ce langage parenthèse pour personnaliser son Xemacs, fonctionne pour toutes ses fonctions (définir un autre dictionnaire, changer la couleur des caractères et à peu près tout ce qu'il peu faire... Y a t'il des choses qu'il ne sais pas faire?)

Gimp.

xset

L'utilitaire de préférence d'utilisateur pour X. Vous pouvez changer toutes les sortes de matériel avec ceci. Par exemple, `xset s 600` place l'écran au blanc après dix minutes

xsetroot

Changer la couleur de votre dessus de bureau. Si vous avez un programme de sélecteur de couleur comme le **xcolorsel** installé, essayer une commande comme le `xsetroot -solid `xcolorsel`` plein pour sélectionner une couleur et pour placer cette couleur sur votre dessus de bureau.

xwininfo

Vous pouvez exécuter ce programme et cliquer dessus n'importe quelle fenêtre pour un bon nombre d'informations utiles sur elle.

The Intellimouse

Vous pouvez employer l'Intellimouse dans X avec un grand beaucoup d'applications. Il y a une excellente page de ressource située au: <http://www.inria.fr/koala/colas/mouse-wheel-scroll/>.

10. Utilisation Avancée de X

Il y a beaucoup à apprendre au sujet de X, beaucoup d'informations enterrés dans la terre insondable des pages du man, beaucoup de choses que la plupart des gens ne prennent pas la peine de lire. Certains toutefois si il sont vraiment importants, (N.D.T.: imaginez que même eux, ne les lises pas toujours complètement, alors qu'ils sont dans leur langues, alors on peut concevoir que la plupart des francophones ne traduisent que l'option ou l'argument qu'ils ont besoin, et pourtant) pourtant il est facile de faire quelques erreurs, et là fatalement se coincent vraiment complètement. Il peut être beaucoup plus difficile d'établir un programme dans X, ou de vérifier la sécurité dans X, ou beaucoup d'autres choses, parce que X et toutes les choses qui fonctionnent sur lui sont si vastes. Cette section est une tentative pour traiter certaines des situations plus avancées et plus embrouillantes qui peuvent surgir. (ndt: Cette traduction l'ai déjà un peu embrouillante, à lire juste pour ce faire une idée, les passage important sont traités avec plus de rigueur et de bon sens.)

10.1 Bibliothèques et Compilation d'Applications X

Tôt ou tard vous devrez traiter des applications de compilation de vous même. Plus tard, si vous installiez une agréable distribution de linux, et êtes heureux avec ce que vous avez, vous verrez que bientôt, surtout si vous êtes le genre de personne qui aime bricoler et installer. Vous serez tenté de faire des mises à jours ou autres transformations qui appelle une connaissance de la compilation et des librairies de X. Rappelez vous, que c'est un privilège, pas un droit, aussi prenez du plaisir avec lui!

D'abord, quelques indicateurs sur la compilation des programmes X. De nombreuses nouvelles applications, les applications de GNU en particulier, viennent avec un script dans le répertoire racine appelé **configure**. Ceci suppose naturellement que vous avez extrait le fichier et êtes dans le répertoire. Ce programme devra être exécuté comme `./configure`, et détectera automatiquement beaucoup de choses au sujet de votre système. Ensuite, en exécutant simplement `make` et peut-être en devenant `root` et exécutant `make install` vous obtiendrez la mise à jours du programme sur votre système.

Vous pouvez devoir faire peu un bricolage si vous n'avez pas un script **configure** disponible. Beaucoup de programmes de X exigent de vous d'exécuter un programme pour faire le **Makefile**, appelé le `xmkmf`. Si vous ne voyez pas un **Makefile** dans votre répertoire, ceci fonctionnera parfois, et générera une configuration appropriée pour vous, et vous serez prêt pour `make` à partir de là.

Parfois vous aurez exécutez le script de **configure**, et avez été avertis que vous manquez d'une bibliothèque qui serait utile ou nécessaire pour employer correctement cette application. Par exemple, j'ai récemment installé l'application de **xscreensaver**, et ai constaté qu'elle soutiendrait plusieurs modes 3d si la bibliothèque Mesa était installée. Si vous exécutez dans cette situation et voulez cette bibliothèque installée, la première chose que vous devriez faire est d'examiner le CD ou de mettre en fonction de moyens pour voir dans votre distribution Linux si vous avez la bibliothèque dessus. Cela peut vous éviter un bon nombre d'ennuis en essayant de compiler la bibliothèque.

Une fois que vous obtenez une bibliothèque compilée et installé et prête pour l'emploi, vous pouvez retourner au répertoire à partir d'ou été installé votre programme X, enlevez le fichier `config.cache`, et exécutez le script de **configure** encore. Si tout va bien il trouvera la bibliothèque que vous avez installée. Parfois non, et vous devez bricoler pour faire en sorte qu'il fonctionne. Ceci se produit de temps en temps avec des bibliothèques que vous téléchargé hors du net, celles-ci ne soutiennent pas en juste proportion la norme de

système de fichiers de Linux (la plupart fonctionne, cependant).

10.2 Sécurité avec X Les Bases

Il a souvent été dit que X a un modèle très simple de sécurité: Tous ou rien. Ce n'est pas beaucoup d'une exagération du tout. X peut être configuré pour utiliser la sécurité légèrement sophistiquée, par l'intermédiaire du chiffrage, mais c'est au delà de la portée de ce HOWTO (pour le moment). Il est supposé que l'utilisateur n'emploie aucun chiffrage pour cette discussion.

Tout d'abord, vous devriez essayer et suivre quelques règles simples quand vous compilez des programmes pour X (ou pour toute raison, vraiment). Essayer de ne pas devenir root plus que nécessaire. Configurer vos programmes en tant qu'utilisateur normal avec le `./configure & & faire` (ou simplement `make` s'il y a aucun script de configuration), et puis publier la commande simple de l'installer comme root par saisie `su -c "make install"`, et alors taper dedans le mot de passe de root au prompt. Cette méthode vous ne devrai pas être entrée comme root, vous pouvez juste publier une simple commande, et vous êtes de nouveau sous votre normal d'utilisateur au prompt. C'est une manière futée et sûre d'installer le logiciel.

La prochaine chose à penser est l'exécution du logiciel X comme root. Se rendre compte que X est plus ou moins en soi peu sûr, et si votre souci primaire pour une boîte particulière Linux est la sécurité, vous ne voudrez vraiment ne pas installer X du tout! Après avoir dit cela, quelques gens veulent exécuter des programmes de fantaisie agréables de configuration ou les outils de gestion de package en X. Je ne recommande pas de commencer X comme root. C'est pas une bonne idée. Il y a beaucoup, des manières bien meilleures de faire ces choses!

Si vous voulez exécuter une application X comme root, l'ouverture juste sous votre compte normal d'utilisateur, et le lancent de là. Comme j'ai mentionné ci-dessus, vous ne voulez pas être entrés comme root plus qu'absolument nécessaire. L'utilisateur root a la capacité de tout faire au sujet de n'importe quoi sur le système, incluant environ un million de manières de le détruire complètement. Aller simplement à votre xterm ou à une telle application et tapez dedans une commande telle que `su -c "glint -display :0.0"` pour lancer la fenêtre sur votre bureau. Maintenant vous avez un fonctionnement simple de fenêtre comme root, et pas le dessus de bureau entier. Vous pourriez également vouloir considérer d'utiliser le nom complet au programme que vous exécutez comme root, pour la sécurité supplémentaire.

Tomasz Motylewski offre également le bout suivant pour des amoureux de **ssh** et d'**ssh-agent**. Si vous mettiez le suivant dans vos fichier `/usr/X11R6/lib/xinit/Xclients` file:

```
eval "exec ssh-agent fvwm${FVWMVER} ${FVWMOPTIONS}" >
"$HOME"/.FVWM${FVWMVER}-errors 2>&
```

C'est-à-dire, remplacer votre commande standard de commande `fvwm${FVWMVER} ${FVWMOPTIONS}` par le précéder. Ce doit démarrer votre gestionnaire de fenêtre en tant qu'enfant d'**ssh-agent**. Alors vous devriez essayer `ssh-add < /dev/null` à partir d'une session **xterm** (si vous avait votre identité clé RSA key dans le fichier `./ssh/identity`). Si sa travail pour vous alors vous devrez rechercher que vous êtes bien authentifié dans toutes vos fenêtre X.

Il convient noter que c'est une suggestion du lecteur, et l'auteur n'a aucune expérience avec le ssh présentement, ainsi procéder à vos propres risques!

10.3 Plus Au Sujet De l'Autorité De X

Les mystères de l'authentification d'utilisateur dans X sont un exemple typique de problèmes de sécurité beaucoup d'utilisateurs que la rencontre avec X. Tomasz Motylewski relie l'histoire suivante, qui résume la situation tout à fait bien.

le ``... dans une session par défaut la configuration de X par XFree86 doit commencé par le ``startx '' accepte des commandes à partir de chacun se reliant à partir du localhost. Si vous démarrez X comme utilisateur 1 et vous avez utilisateur 2 loggé à distance, utilisateur 2 a plein accès à votre clavier et au dessus du bureau (une fois que mon ami jouait une plaisanterie sur moi et mettait dans sa commande cron vidant mon image de bureau de X et lui envoyant chaque heure). Je ne l'ai pas noté pendant 2 semaines! ''

D'accord ce n'est pas exactement une bonne chose, mais malheureusement beaucoup de lecteurs ont ce problème exact. Tomasz continue pour préciser que si vous employez le **xdm** pour ouvrir une session (comme détaillé plus tôt dans ce HOWTO), plutôt que le **startx** courant, ce ne sera pas un problème parce que l'accès est commandé par le programme de **xauth**. En publiant la commande de **xhost**, vous pouvez voir qui est autorisé à l'accès à votre session de X. N'importe quel utilisateur des centres serveurs qui sont énumérés dans le rendement de la commande de **xhost** est autorisé à plein accès pour votre écran et clavier.

Si vous voulez commencer votre serveur de X dans un mode bloquée de la ligne de commande, vous devrez faire les modifications suivantes à votre configuration. D'abord, modifier le **startx** (probablement à */usr/X11R6/bin/startx*) par changement de la dernière ligne `xinit $clientargs -- $serverargs to lire exec xinit $clientargs -- $serverargs -auth ~/.Xauthority.`

Alors aurai besoin aussi d'éditer votre **.xinitrc** ou votre système étendu */usr/X11R6/lib/xinit/xinitrc* (peut importe lequel vous êtes actuellement entrain d'utiliser, probablement le premier dans votre répertoire home), par l'ajout de ça au début du fichier:

```
# si Xserver a ete lancer avec option "-auth ~/.Xauthority"
cela
# empechera les autres utilisateurs sur votre machine de se
connecter
# a votre X server à moins que vous le permettiez
explicitement en
# utilisant xhost +host ou donner leurs votre fichier
~/.Xauthority.
xauth generate $DISPLAY . trusted
```

Toujours être sur et exécuter **xhost** pour contrôler la sécurité que vous avez configuré, pour être sûr que tout travaille correctement.

[Suivant](#) | [Précédant](#) | [Retour sujet](#) [Suivant](#) | [Précédant](#) | [Retour sujet](#)

11. Bibliographie et Autre Ressources

Il y a un bon nombre d'information sur X Window à trouver, sur votre ordinateur en ce moment, sur l'Internet, et en quelques excellents livres disponibles dans votre librairie préféré. Essayez certaines des ressources suivantes, elles ont été extrêmement utiles à l'auteur, et peuvent aussi vous être extrêmement utiles.

- Examiner le website [Linux Documentation Project](#) pour une plus ample documentation sur Linux, X, et articles relatifs.
 - <http://www.x11.org/> est une sorte de chambre de compensation pour toutes les choses X.
 - <http://www.themes.org/> est le siège social pour Themes pour différents gestionnaires de fenêtres.
 - Le site Web du Consortium de X est <http://www.x.org/>... ou peut-être est-il déplacé à <http://www.opengroup.com/>.
 - XFree86 peut être trouvé à <http://www.xfree86.org/>.
 - La série d'O'Reilly sur les X Window! Visitez le <http://www.ora.com/> les livres définitifs sur X.
 - Beaucoup plus d'information sur l'utilisation des TrueType avec X au <http://www.freetype.org/>.
 - Noah Roberts a écrit un excellent papier sur des ressources de X qui est une sorte de cours d'instruction plus détaillé au concept de ressource. Vous pouvez le trouver au <http://home.sprintmail.com/~fract/>.
 - Les pages du man pour **X**, le **xterm**, **XFree86**, et pour tous les autres clients, vous en trouverez vous même souvent l'utilité, sont aussi utiles les informations compressées fortement recommandées. Comme les maintes fois répétés RTFM.
 - Il y a une Mini HOWTO pour les application d'X à distances (**Remote X Apps** MINI-HOWTO) qui est très utile dans l'appréhension extérieure de la façon d'exécuter les clients locaux et à distance avec X.
 - Et spécialement pour les francophones les sites toujours bien garnis de: [http://www.freenix.org/unix/linux/HOWTO/...](http://www.freenix.org/unix/linux/HOWTO/) ainsi qu'à linux-france.org et aussi sur les autres ...
-

Suivant | [Précédant](#) | [Retour sujet](#)