

Cygwin/設定・運用編

<http://www.kisnet.or.jp/~kanou/index.php?Cygwin%2F%E8%A8%AD%E5%AE%9A%E3%83%BB%E9%81%8B%E7%94%A8%E7%B7%A8>

Cygwin/設定・運用編

XFree86のインストールが終わったので、いよいよLinuxサーバに接続するための設定を行ってこよう。接続の設定とともに、キーボードやフォントの設定を変更して実用度を向上させることも忘れてはならない。

- [Cygwin/XFree86の設定](#)
 - [Cygwin/XFree86のキアサインを106キーボードに](#)
- [Linuxサーバへのアクセス](#)
 - [sshで安全にアクセスする](#)
- [\[XDMCP\]によるグラフィカルログイン](#)
 - [\[XDMCP\]のためのLinuxの設定](#)
 - [\[XDMCP\]によるリモートセッションの開始](#)
 - [フォントサーバでLinuxのフォントを使う](#)
- [Cygwin/XFree86でウィンドウマネージャを動かそう](#)
 - [軽量ウィンドウマネージャ「Blackbox」](#)
 - [定番ウィンドウマネージャ「Window Maker」](#)
- [Appendix](#)

Cygwin/XFree86の設定 [±]

startxwin.batやstartxwin.shを見るとCygwin/XFree86の設定がよく分かります。Xサーバを起動するのは**XWin.exe**です。Xサーバを起動するだけなら、次のようにします。

```
$ XWin -screen 0 800x600x256
```

ただし、これだと何も表示されていないウィンドウが現れるだけです。続けて、xtermでbashを起動したり、ウィンドウマネージャ(デフォルトはtwm)を起動する必要があります。具体的には、以下のようなシェルスクリプトを作成し、Cygwinからこれを実行すればいいでしょう。

```
#!/bin/sh
XWin -screen 0 800x600x256 &
rxvt -e bash &
twm &
```

2行目 800×600ドット、256色モードでXを起動

3行目 rxvtでbashを起動

4行目 ウィンドウマネージャをtwmに設定

これで、一応Xでbashが使えるようになります。XWin.exeのオプションは多岐にわたっているので、詳しくは**man XWin**を見たり、**XWin -h**を実行してヘルプを参照してください。

Cygwin/XFree86のキアサインを106キーボードに [±]

Cygwin/XFree86でbashを起動すると、キー配列が101キーボードになってしまい、「&」や「~」といった記号が思うように入力できなくなります。通常のLinuxでは、kbdconfigなどのツールを使って106キーボードに設定すればいいのですが、Cygwin/XFree86ではXの起動プロセスが異なるため、この方法は使えません。

Cygwin/XFree86のWebサイトにあるFAQには、非USキーボードに設定する方法が載っています。それによると、xmodmapコマンドでxmodmap.jpを読み込めばいいと書いてあるのですが、**xmodmap.jp**がどこにも存在しません。

そこで、xmodmap.jpを作成しましょう。作成の方法は簡単です。正常に動作しているLinux(筆者はTurbolinux 6.0 Workstationを使用)で、以下のコマンドを実行するだけです。

```
$ xmodmap -pke > xmodmap.jp
```

できあがったxmodmap.jpをCygwin/XFree86が動作しているPCにコピーして、bashで以下のコマンドを実行すれば**106キーボード**の配列になります。

```
$ xmodmap /tmp/xmodmap.jp
```

例では、便宜的にxmodmap.jpを/tmpに置いた

Xの起動にstartxコマンドを使う場合は、ホームディレクトリに置いたxmodmap.jpを**Xmodmap**という名前にリネームすれば自動的に読み込まれます。ただし、環境によってはstartxは動作が不安定になるようです(コラム:XFree86のコンパイル

Cygwin用に用意されたバイナリでstartxの挙動が不安定な場合は、XFree86 4.1.0をソースからコンパイルすることで解決する可能性がある。ただし、ダウンロードやコンパイルには時間がかかるので覚悟が必要だ。

XFree86をコンパイルする場合は、本家(<http://www.xfree86.org/>)のFTPサイトにあるソースを利用する(<ftp://ftp.xfree86.org/pub/XFree86/4.1.0/source/>)。ここにある

```
X410src-1.tgz
X410src-2.tgz
X410src-3.tgz
```

の3ファイル(合計約54.1Mbytes)をダウンロードし、/tmpなどで展開する。するとxcというディレクトリができるので、ここに移動して

```
$ make World
$ make install
```

を実行する。これでXFree86のインストールは完了だ。ちなみに、make Worldとmake installが終わるのに、Pentium III-766MHzのマシンでも3時間強を要した(編集局)。)。その場合は、シェルスクリプト(例えばstartxwin.sh)の中でxmodmapを実行してください。

```
#!/bin/sh
export DISPLAY=127.0.0.1:0.0
PATH=/usr/X11R6/bin:$PATH

# Cleanup from last run.
rm -rf /tmp/.X11-unix

XWin -screen 0 800x600x256 &
twm &
xmodmap xmodmap.jp &
xsetroot -solid aquamarine4
xterm -n xterm-1 -j -ls -sb -sl 500 -rightbar -geometry +0+0 -e bash &
exit
```

Linuxが動作する環境をお持ちでない方のために、筆者が作成したxmodmap.jpを公開します。本来、xmodmap.jpには変更したいキーだけを定義すればいいはずですが、時間の関係で全キーの配列が定義されたままになっています。

Linuxサーバへのアクセス [±]

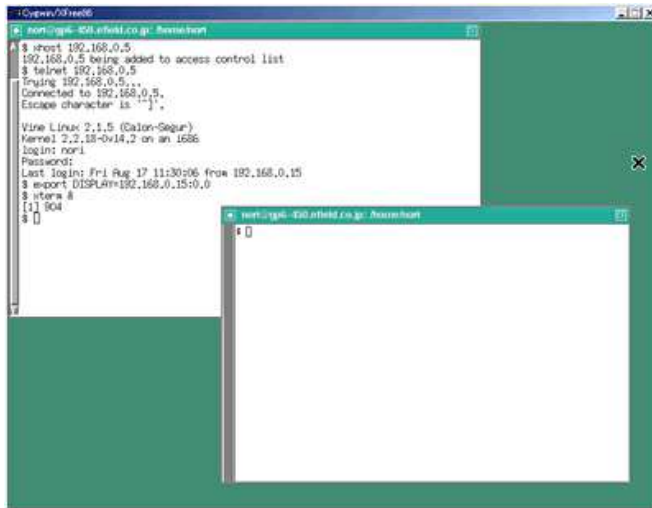
さて、いよいよ主目的の1つであるLinuxサーバへのアクセスです。ここでは、192.168.0.15というIPアドレスを持つCygwin/XFree86から、192.168.0.5というIPアドレスを持つLinuxのサーバにアクセスしてみます。

```
$ xhost 192.168.0.5 ←Xサーバ (Cygwin/XFree86) への接続を許可するホストを登録
192.168.0.5 being added to access control list
$ telnet 192.168.0.5 ←Linuxサーバに接続
Trying 192.168.0.5...
Connected to 192.168.0.5.
Escape character is '^]'.

Vine Linux 2.1.5 (Calon-Segur)
Kernel 2.2.18-0v14.2 on an i686
login: nori
Password:
Last login: Fri Aug 17 11:30:06 from 192.168.0.15
$ export DISPLAY=192.168.0.15:0.0 ←Linuxサーバからの出力 (表示) をCygwin/XFree86に送るように設定
```

```
$ xterm & ←xtermをバックグラウンドで起動
```

Linuxサーバにtelnetでログインして、出力先をCygwin/XFree86に設定します。そしてxtermを起動すると、Linuxサーバ上で起動したxtermの画面がCygwin/XFree86上に表示されます。これで、Linuxサーバの操作をすべてCygwin/XFree86から行えることになります。



画面10 左上のウィンドウがCygwin/XFree86のrxvtで、左下のウィンドウがLinuxサーバのxterm (画像をクリックすると拡大表示します)

残念なことに、現在のCygwin/XFree86はWindowsのIMEから漢字を入力することができません。これについては、今後のバージョンアップに期待するしかなさそうです。

sshで安全にアクセスする ¹

上では、Cygwin/XFree86からサーバにtelnetでアクセスする方法を紹介しました。LANの内部で使用する分にはtelnetでも構いませんが、外部からアクセスするにはセキュリティの面で心配です。そんなときはsshを利用します。CygwinにはOpenSSHが標準でインストールされているので、そのままsshを使用することができます。

ここでは、デフォルトの設定で使用できるホストベース認証を行います。ホストベース認証によるログインでもパスワードは暗号化して送信されるので、パスワードが漏えいする心配はありません。

sshでアクセスするには、上記の「telnet 192.168.0.5」の部分を「ssh 192.168.0.5」に変えるだけです。もちろんサーバ側の設定も必要ですが、最近のディストリビューションであれば最低限の設定はしてあるので、そのままsshでアクセスできます。Red Hat系ならばntsysvコマンドで、Turbolinuxの場合はturboserviceコマンドでsshdが起動しているかどうかを調べ、起動していない場合は起動してください。

ではsshによる接続の手順です。xhostでホストを登録し、sshでサーバにアクセスします。

```
$ xhost 192.168.0.5 ←サーバ (Cygwin/XFree86) への接続を許可するホストを登録
$ ssh 192.168.0.5 ←sshで192.168.0.5に接続
The authenticity of host '192.168.0.5 (192.168.0.5)' can't be established.
DSA key fingerprint is 5a:69:c2:85:16:1d:44:75:80:ff:92:b3:14:0a:ca:45.
Are you sure you want to continue connecting (yes/no)? yes ←「yes」と入力
Warning: Permanently added '192.168.0.5' (DSA) to the list of known hosts.
noriyu-k@192.168.0.5's password: ←サーバーのパスワードを入力
Last login: Tue Aug 21 17:26:57 2001 from localhost.localdomain
Welcome to Turbolinux.
$ ←サーバーのbashが表示される
$ export DISPLAY=192.168.0.15:0.0 ←Linuxサーバからの出力 (表示) をCygwin/XFree86に送るように設定
$ xterm & ←xtermをバックグラウンドで起動
```

[XDMCP]によるグラフィカルログイン ¹

[XDMCP]のためのLinuxの設定 ¹

XDMCP(X Display Manager Control Protocol)によって、ログインをはじめとするLinuxサーバ上のすべての操作をCygwin/XFree86から行えば、Linux上のかな漢字変換システムを使って漢字を入力することができます。

ただし、キー入力と画面表示以外は全部Linuxサーバが行うことになり、Cygwin/XFree86とLinuxサーバ間のトラフィックも増えるので、動作速度はかなり遅くなります。

また、この方法を使うには条件が2つあります。1つは、Linuxサーバがグラフィカルログインの設定(ランレベル5)になっていることです。もう1つは、Linuxサーバのディスプレイマネージャ(xdm、gdm、wdmなど。ディストリビューションによって異なる)が、外部からのアクセスを許可していることです。

ランレベルを5にして起動するのは簡単です。一時的にそうしたいのであれば、LILOのプロンプトで

```
LILO: linux 5
```

とするか、ログイン後に

```
# telinit 5
```

を実行すればいいのです。常にグラフィカルログイン画面にしたいなら、/etc/inittabを以下のように編集します。

```
id:5:initdefault: ←3を5に変更
```

Linuxサーバのディスプレイマネージャが外部からのアクセスを許可するには、コンフィグレーションファイルを変更します。問題は、ディストリビューションによって使用しているディスプレイマネージャ(gdm, xdmなど)が異なることです。

gdmの場合は、**/usr/X11/gdm/gdm.conf**を編集します。

```
[xdmcp]
Enable=1 ←0を1に変更
Port=177 ←177でないときは177に変更
```

もし、これでもグラフィカルログイン画面が表示されないときは、以下の変更も加えてください。gdmを使用しているディストリビューションには、Red HatやTurbolinuxがあります。

```
id:5:initdefault: ←3を5に変更
```

xdmの場合は、/usr/X11/xdm/xdm-configを編集します。

```
DisplayManager.requestPort: 177 ←0を177に変更
```

Vine Linuxはwdmを使用していますが、デフォルトでXDMCPを許可しているので、変更は必要ありません。

[XDMCP]によるリモートセッションの開始 [↑]

Linuxサーバ側の設定が終わったら、後はWindows から接続するだけです。

XDMCPによるリモートセッションを行うには、XWin.exeで以下のようなオプションを付加します。-queryにはLinuxサーバのIPアドレスを、-fromはCygwin/XFree86のIPアドレスを指定します。

```
$ XWin.exe -screen 0 800x600x256 -query 192.168.0.12 -from 192.168.0.11
```

各設定を正しく行っていれば、Linuxのグラフィカルログイン画面が表示されます。

画面11 Cygwin/XFree86に現れたVine Linux 2.1.5のグラフィカルログイン画面



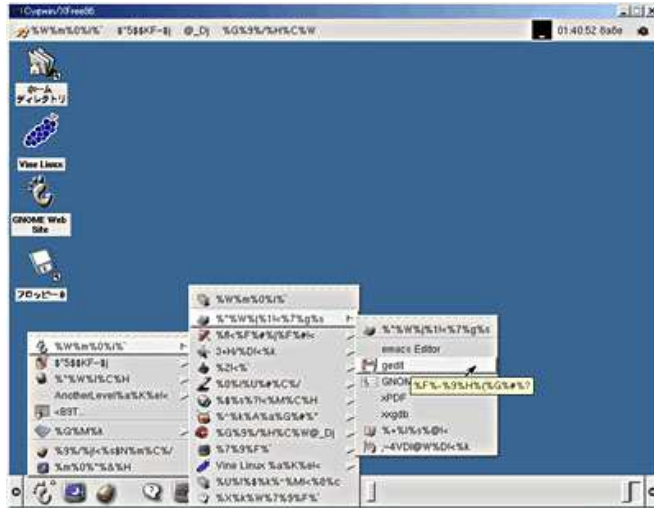
画面11 Cygwin/XFree86に現れたVine Linux 2.1.5のグラフィカルログイン画面 (画像をクリックすると拡大表示します)

ここでLinuxサーバのユーザーアカウントとパスワードを入力すれば、Cygwin/XFree86上にLinuxサーバのX画面が表示されます。

フォントサーバでLinuxのフォントを使う¹

さて、上でサラリとVine Linux 2.1.5のグラフィカルログイン画面が現れて、Cygwin/XFree86上にLinuxサーバのX画面が表示されると書きましたが、実はデフォルトの設定では漢字が化けてしまって使い物になりません。

画面12 Vine LinuxのGNOMEが表示されたが、漢字が化けていて使えない



画面12 Vine LinuxのGNOMEが表示されたが、漢字が化けていて使えない(画像をクリックすると拡大表示します)

これを防ぐには、フォントサーバの機能を使います。つまり、Cygwin/XFree86の表示をLinuxのフォントを使って行うのです。

まず、Linux側の設定を変更しましょう。Vine Linuxにrootでログインして、`/etc/rc.d/init.d/xfs`を開きます。以下の行を捜して、「-1」を「7100」に変更します。同じ内容の行が2箇所あるので、両方を修正してください。

```
daemon xfs -droppriv -daemon -port 7100 ←-1を7100に変更
```

次に、`/etc/X11/XF86Config`を開いて、

```
FontPath "unix/:-1"
```

という行を以下のように変更します。

```
FontPath "tcp/localhost:7100"
```

修正が終わったら、Linuxを再起動します。以上で、設定は終了です。

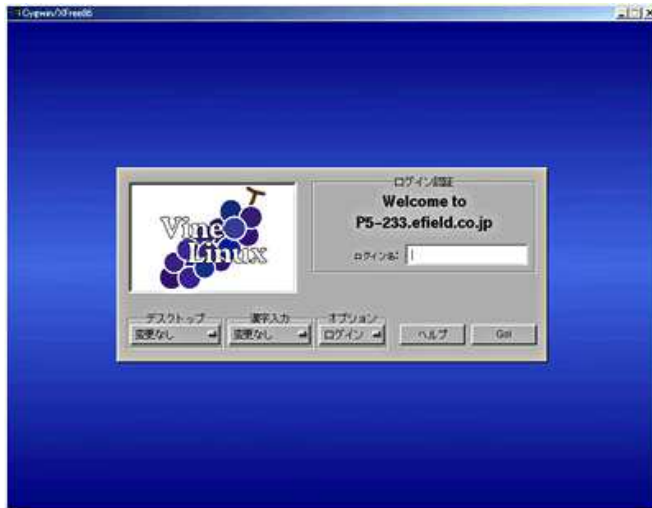
Linuxのフォントを使用するには、Cygwin/XFree86起動時に`-fp`オプションでFontPath²を指定します。書式は「`-fp フォントパス`」となります。例えば、192.168.0.5のLinuxに接続する場合は、

- `fp tcp/192.168.0.5:7100`

とします。例のごとく、シェルスクリプトにしておくとう便利です。

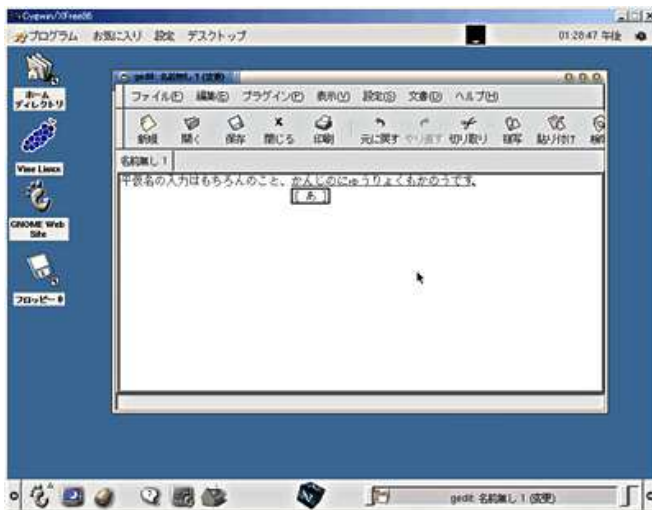
```
XWin -screen 0 800x600x256 -query 192.168.0.5 -from 192.168.0.15 -fp tcp/192.168.0.5:7100 &
```

画面13 フォントサーバの機能を使用すると、Vine Linuxとまったく同じ表示になる(画像をクリックすると拡大表示します)



画面13 フォントサーバの機能を使用すると、Vine Linuxとまったく同じ表示になる(画像をクリックすると拡大表示します)

画面14 XDMCPでLinuxサーバにログインするとGNOMEが起動する。Linuxのかな漢字変換機能を使って日本語を入力することもできる(画像をクリックすると拡大表示します)



画面14 XDMCPでLinuxサーバにログインするとGNOMEが起動する。Linuxのかな漢字変換機能を使って日本語を入力することもできる(画像をクリックすると拡大表示します)

Cygwin/XFree86でウィンドウマネージャを動かそう [↑]

前ページまでの解説で、CygwinによるXサーバの構築という目的は達成された。ここまででも十分なのだが、せっかくなのでもう少しCygwinで遊んでみよう。

Xサーバとして利用するなら、ウィンドウマネージャはデフォルトのtwmでもよいのだが、少々さびしいのも事実だ。そこで、ウィンドウマネージャの変更に挑戦してみたい。前ページでGNOMEの画面があるが、これはLinux上で動いているものをWindowsの画面に表示させているに過ぎない。これから紹介するのは、Windows上で直接ウィンドウマネージャを動かす方法なのだ。

軽量ウィンドウマネージャ「Blackbox」 [↑]

カスタマイズ性と軽快な動作で人気のウィンドウマネージャがBlackbox(<http://blackbox.alug.org/>)だ。まずはこれをCygwin/XFree86で動かしてみる。Blackboxは、Cygwin/XFree86用のバイナリが用意されている。これをダウンロードして所定のディレクトリに配置すれば動くので非常に手軽だ。

バイナリは<http://cygwin.com/mirrors.html>からたどれるいずれかのミラーサイトの/**xfree/contrib/**にある。ここにある**blackbox-0.51.3.1-bin.tar.bz2**をダウンロードしよう。ダウンロードしたらCygwinを起動して、ファイルを適当なディレクトリ(例では/tmp)に移動。これを以下のコマンドで展開する。

```
$ bzip2 -dc blackbox-0.51.3.1-bin.tar.bz2 | tar xvf -
```


これだけでファイルがそれぞれのディレクトリに配置される。つまりインストール完了だ。後はBlackboxをウィンドウマネージャに指定するだけである。

startxでXを起動しているなら、ホームディレクトリの.xinitrcを編集する。

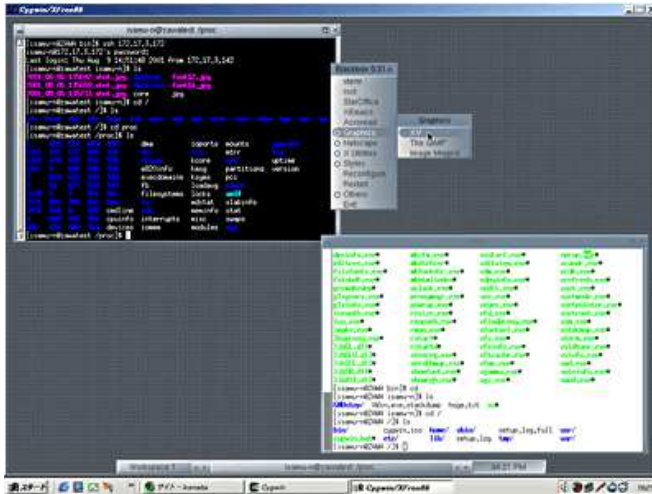
```
twm &
```

という行を探し、ここを

```
blackbox &
```

にすればよい。.xinitrcを修正したらstartxしてみよう。

Cygwin/XFree86上で起動したBlackbox (画像をクリックすると拡大表示します)



Cygwin/XFree86上で起動したBlackbox (画像をクリックすると拡大表示します)

定番ウィンドウマネージャ「Window Maker」¹

ウィンドウマネージャにWindow Makerを使っている人も多いだろう。Windows でも、Cygwin/XFree86を使えば**Window Maker**が動いてしまうのだ。

とはいえ、Cygwin用のバイナリは用意されていない。ではどうするかというと、オリジナルのソースをWindows上でコンパイルするのだ。つまり、Window Maker自体はLinuxなどで動作するものと同一ということである。

まず、Window Makerのソースを<http://www.windowmaker.org/>からダウンロードする。ここでダウンロードすべきは

- libPropList? 0.10.1(libPropList?-latest.tar.gz)
- Window Maker 0.65.0(WindowMaker?-0.65.0.tar.gz)

の2ファイルだ。2001年8月現在、Window Makerの最新版は0.65.1なのだが、このバージョンではmakeに失敗してしまう。今回は時間の関係で原因を追求できなかったため、makeがそのまま通った0.65.0を使うことにした。

ダウンロードしたファイルは例によってCygwinの/tmpに移動して展開する。

ではインストールだ。まずlibPropList?から行う。

```
$ tar xzvf libPropList-latest.tar.gz
```

で展開したら、生成されたlibPropList?-0.10.1ディレクトリに移動する。ここで

```
$ ./configure
$ make
$ make install
```

を実行すればlibPropList?のインストールは完了だ。

続いてWindow Maker本体をインストールする。これも

```
$ tar xzvf WindowMaker-0.65.0.tar.gz
```

でアーカイブを展開し、**WindowMaker?-0.65.0**ディレクトリに移動する。そして、こちらも同じく

```
$ ./configure
$ make
$ make install
```

を実行すればよい。パッチなどは不要だ。

次に、環境設定用スクリプトの**wmaker.inst**を実行する。これは/usr/loca/binにある。

```
$ cd /usr/loca/bin
$ ./wmaker.inst
```

最後に.xinitrcを書き換える。「twm &」あるいは「blackbox &」となっている行を

```
exec wmaker
```

に変更しよう。これでWindow Makerの準備は完了だ。startxすればWindow Makerが起動する。



コンパイルしたWindow Makerを実行 (画像をクリックすると拡大表示します)

これ以外にも、Cygwin用KDEの開発プロジェクトなどもある(<http://sourceforge.net/projects/kde-cygwin/>)。ここまでくると、Cygwinではなく本物のLinuxを使った方がよいのではないかという気もするが、興味のある人は挑戦してみるのも面白いだろう。

Appendix [±]

以下のリストはLinux(Turbolinux 6.0 Workstation)から作成したxmodmap.jpです。これをxmodmap.jp(あるいはXmodmap)というファイルにカット&ペーストするか、以下のリンクからxmodmap.jpをダウンロードしてご利用ください。リンクを右クリックして[対象をファイルに保存](IE)、[リンクを名前を付けて保存](Netscape)すればダウンロードできます。

ダウンロード用xmodmap.jp

keycode 8 =

keycode 9 = Escape

keycode 10 = 1 exclam kana_NU exclamdown

keycode 11 = 2 quotedbl kana_FU oneighth

keycode 12 = 3 numbersign kana_A kana_a

keycode 13 = 4 dollar kana_U kana_u

keycode 14 = 5 percent kana_E kana_e

keycode 15 = 6 ampersand kana_O kana_o

keycode 16 = 7 apostrophe kana_YA kana_ya

keycode 17 = 8 parenleft kana_YU kana_yu
keycode 18 = 9 parenright kana_YO kana_yo
keycode 19 = 0 asciitilde kana_WA kana_WO
keycode 20 = minus equal kana_HO questiondown
keycode 21 = asciicircum asciitilde kana_HE dead_ogonek
keycode 22 = BackSpace?
keycode 23 = Tab ISO_Left_Tab
keycode 24 = q Q kana_TA Greek_OMEGA
keycode 25 = w W kana_TE Lstroke
keycode 26 = e E kana_I kana_i
keycode 27 = r R kana_SU registered
keycode 28 = t T kana_KA Tslash
keycode 29 = y Y kana_N yen
keycode 30 = u U kana_NA uparrow
keycode 31 = i I kana_NI idotless
keycode 32 = o O kana_RA Ooblique
keycode 33 = p P kana_SE THORN
keycode 34 = at grave voicedsound dead_abovering
keycode 35 = bracketleft braceleft semivoicedsound kana_openingbracket
keycode 36 = Return
keycode 37 = Control_L
keycode 38 = a A kana_CHI AE
keycode 39 = s S kana_TO section
keycode 40 = d D kana_SHI ETH
keycode 41 = f F kana_HA ordfeminine
keycode 42 = g G kana_KI ENG
keycode 43 = h H kana_KU Hstroke
keycode 44 = j J kana_MA
keycode 45 = k K kana_NO ampersand
keycode 46 = l L kana_RI Lstroke
keycode 47 = semicolon plus kana_RE dead_doubleacute
keycode 48 = colon asterisk kana_KE dead_caron
keycode 49 = Zenkaku_Hankaku Kanji
keycode 50 = Shift_L
keycode 51 = bracketright braceright kana_MU kana_closingbracket
keycode 52 = z Z kana_TSU kana_tsu
keycode 53 = x X kana_SA greater
keycode 54 = c C kana_SO copyright
keycode 55 = v V kana_HI grave

keycode 56 = b B kana_KO apostrophe
keycode 57 = n N kana_MI
keycode 58 = m M kana_MO masculine
keycode 59 = comma less kana_NE kana_comma
keycode 60 = period greater kana_RU kana_fullstop
keycode 61 = slash question kana_ME kana_conjunctive
keycode 62 = Shift_R
keycode 63 = KP_Multiply
keycode 64 = Alt_L Meta_L
keycode 65 = space
keycode 66 = Caps_Lock Eisu_toggle
keycode 67 = F1
keycode 68 = F2
keycode 69 = F3
keycode 70 = F4
keycode 71 = F5
keycode 72 = F6
keycode 73 = F7
keycode 74 = F8
keycode 75 = F9
keycode 76 = F10
keycode 77 = Num_Lock Pointer_EnableKeys?
keycode 78 = Scroll_Lock
keycode 79 = KP_Home KP_7
keycode 80 = KP_Up KP_8
keycode 81 = KP_Prior KP_9
keycode 82 = KP_Subtract
keycode 83 = KP_Left KP_4
keycode 84 = KP_Begin KP_5
keycode 85 = KP_Right KP_6
keycode 86 = KP_Add
keycode 87 = KP_End KP_1
keycode 88 = KP_Down KP_2
keycode 89 = KP_Next KP_3
keycode 90 = KP_Insert KP_0
keycode 91 = KP_Delete KP_Decimal
keycode 92 =
keycode 93 =
keycode 94 =

keycode 95 = F11
keycode 96 = F12
keycode 97 = Home
keycode 98 = Up
keycode 99 = Prior
keycode 100 = Left
keycode 101 =
keycode 102 = Right
keycode 103 = End
keycode 104 = Down
keycode 105 = Next
keycode 106 = Insert
keycode 107 = Delete
keycode 108 = KP_Enter
keycode 109 = Control_R
keycode 110 = Pause Break
keycode 111 = Print Execute
keycode 112 = KP_Divide
keycode 113 = Alt_R Meta_R
keycode 114 =
keycode 115 =
keycode 116 =
keycode 117 =
keycode 118 =
keycode 119 =
keycode 120 = Hiragana_Katakana Romaji
keycode 121 =
keycode 122 =
keycode 123 = backslash underscore kana_RO
keycode 124 =
keycode 125 =
keycode 126 =
keycode 127 =
keycode 128 =
keycode 129 = Henkan_Mode Mode_switch
keycode 130 =
keycode 131 = Muhenkan
keycode 132 =
keycode 133 = backslash bar prolongedsound

keycode 134 =

リスト xmodmap.jp

コラム: XFree86のコンパイル

Cygwin用に用意されたバイナリでstartxの挙動が不安定な場合は、XFree86 4.1.0をソースからコンパイルすることで解決する可能性がある。ただし、ダウンロードやコンパイルには時間がかかるので覚悟が必要だ。

XFree86をコンパイルする場合は、本家(<http://www.xfree86.org/>)のFTPサイトにあるソースを利用する(<ftp://ftp.xfree86.org/pub/XFree86/4.1.0/source/>)。ここにある

- X410src-1.tgz
- X410src-2.tgz
- X410src-3.tgz

の3ファイル(合計約54.1Mbytes)をダウンロードし、/tmpなどで展開する。するとxcというディレクトリができるので、ここに移動して

```
$ make World
$ make install
```

を実行する。これでXFree86のインストールは完了だ。ちなみに、make Worldとmake installが終わるのに、Pentium III-766MHzのマシンでも3時間強を要した。

Last-modified: 2016-03-06 (日) 19:01:39 (0m)

Site admin: [安定版の最新バージョンは PukiWiki 1.5.0 です \(PHP5.5対応\)](#)

PukiWiki 1.5.0 Copyright © 2001-2006 [PukiWiki Developers Team](#). License is [GPL](#).
Based on "PukiWiki" 1.3 by [yu-ji](#). Powered by PHP 5.3.3. HTML convert time: 0.216 sec.