

本稿は [Linux Japan 誌](#) 1999 年 7 月号に掲載された記事に補筆修正したものです。

GUI マウントツール

先日、家族と吉祥寺の西武ロフトに行きました。私、文房具が大好きでして、珍しい文房具を見つけては、ああでもないこうでもない、重箱の隅をつつくようなチェックを入れて悦にいつているのです。あれっ、この姿ってどこかで見たような？。そうなんです、私がおじさんです... じゃない(さっむー)、これはまさに X11 のデスクトップの小物をいじって楽しんでいるのと同じじゃありませんか。というわけで、今週も小間物屋のお開きでございます。

勤め先と自宅でデータを共有する方法を考えましょう。勤め先が外部からの telnet や ftp を許可しているなら、自宅からプロバイダー経由で勤め先のホストに入ってファイルを持ってくるのがスマートでしょう。しかし、ファイルが MB オーダーになると通信費用も馬鹿になりません。結局、リムーバブルメディアを持ち運ぶことになるのではないのでしょうか。私は、この頃 PCMCIA の 20MB フラッシュメモリーを愛用しています。もともと、デジタルカメラの記憶媒体に使うつもりだったのですが、軽くて早くて埃にも強くすっかり気に入ってしまいました。皆さんはどうですか？一般的には FD, MO, ZIP, PD でしょうね。

デバイスのマウント：mount/umount

ここで、このような記憶媒体を使う手順をおさらいしましょう。デバイスが認識されたなら、

- 媒体をフォーマットし、ファイルシステムを作成する
- root 権限でマウントできるかどうか試してみる
- user 権限でマウントできるように /etc/fstab に記述を追加する

となるでしょう。少し具体的に話すと、

mke2fs

物理フォーマットは終わっている場合が多いので、HD として扱えるものは fdisk でパーティションをきってから、また FD ではそのまま、mke2fs すれば ext2 ファイルシステムができます。mkdosfs なら MSDOS や FAT32 ファイルシステムを作成できます。もちろん root 権限が必要です。例えば、

```
mke2fs /dev/fd0u1440 ..... フロッピー  
mke2fs /dev/sda4 ..... SCSI 接続の zip
```

などとします。この他に FD を MSDOS でフォーマットするコマンド mformat (mtools) も覚えておいて損はないでしょう。近頃は、オプション-F で FAT32 も作成できるようです (long name が使えますね)。

お試し mount

続いて root 権限でマウントが成功するかどうかを確かめましょう。その前に、mkdir でディレクトリ(マウントポイント)を作っておく必要があります。例えば

```
/mnt/floppy /floppy /fd0 ..... フロッピー  
/mnt/cdrom /cdrom ..... CD-ROM  
/mnt/card /card ..... フラッシュメモリ
```

など、お好みでどうぞ。そうして、メディアをドライブに入れて

```
mount /dev/hde1 /card ..... (*)  
mount /dev/sda4 /mnt/zip
```

などと実行してください。マウントの成功は df で確かめられます。さてこの時点でデバイスのパーミッションを 777 に変更しておくといいでしょう。例えば次のように実行します。

```
chmod 777 /card
```

オーナーは root のままで結構。なぜなら、オーナーを自分のログイン名にしたとしても、勤め先では(ログイン名は同じかもしれませんが)異なる uid が割り当てられている可能性が高いので、同じユーザーと認識されず全く意味がないからです。

/etc/fstab の編集

最後は、一般ユーザーでもマウントが実行できるように /etc/fstab を編集しましょう。デバイス名、ディレクトリ(マウントポイント)名、ファイルシステム名、オプションの順に空白で区切って並べます。例えば

```
/dev/cdrom /cdrom iso9660 ro,user,noauto  
/dev/fd0 /floppy auto rw,user,noauto,exec  
/dev/hde1 /card auto rw,user,noauto,exec
```

のように記述を追加します。オプションではユーザー権限でマウントができるようにするために、必ず user を指定しましょう。ファイルシステム名の auto は、ファイルシステムの種類を自動判別してマウントするオプションです。

```
cat /proc/filesystem
```

で自動判別の対象となるファイルシステムの一覧ができます。

以上、`/etc/fstab` を編集した後は、ユーザー権限で以下のようにコマンドを発行してリムーバブルメディアのマウントが可能となります。

```
mount /card
```

このコマンド文「お試し mount」の(*)とは異なっている点に注意してください。(*)の形式でマウントするには root 権限が必要です。

マウントの GUI フロントエンド

デバイスはマウントしないとファイルシステムに組み込まれませんから、`mount` コマンドの実行は必要不可欠です。ただ、すでにマウント済みのデバイスに対して更に `mount` を実行すると、文句を言われるのがちょっと苦々しいです(&>/dev/null するのは無しです)。そこで `mount` の状態を表示する小物が欲しくなります。この手のものには `xvmount`、`xmount`、`xmmount`、`wmmount` などがあり、状態の表示とボタン操作による `mount/umount` の実行をしてくれます。

`xvmount`^[1]^{W³}

数年前には、私の知る限りこれしかなかったような気がします。作者は Tom (uhl@fh-heilbronn.de) 氏と README に記載されています。現在 Xview の環境が標準ではインストールされないのでコンパイルもままなりません。しかし、この `xvmount` を真似て `xmount` や `xmmount` が開発されたので、功績大というところでしょう。

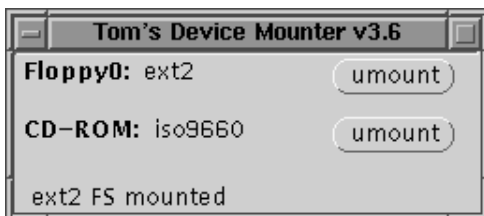


図 1 懐かしい `xvmount`

`xmount`^[2]^{W³}

Kidong Lee (kidong@nownuri.nowcom.co.kr) 氏が、`xvmount` を参考にして、最も標準的な X の開発ベースである Athena ウィジェットで `xmount` を作成しました。README の説明によると、2.0.0 が不安定なため `auto-detect` の機能は削除したとのことです。`/etc/fstab` に良く似たコンフィグレーションファイル `/etc/xmounttab` を使います。オーナーを `root` にし、`setuid` しなければならないところも `xvmount` 譲りです。

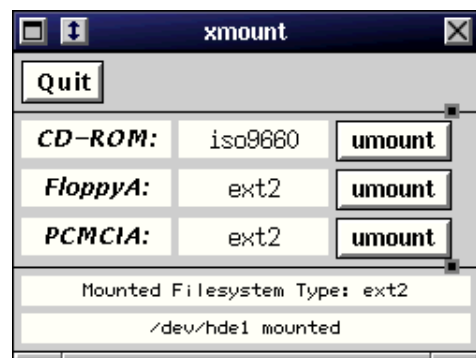


図 2 Athena Widget ベースの `xmount`

`xmmount`^[3]^{W³}

harald.stauss@rz.hu-berlin.de 氏はモチーフウィジェット (LessTif でも可能) で `xmmount` を開発しました。やはり、お手本は `xvmount` だそうです。minix, ext, ext2, msdos, xiafs, iso9660 を認識し、`auto-detect` も大丈夫。`/etc/xmmounttab` を使います (綴を間違えそうですね)。`setuid` も同様に必要 (悪?) です。

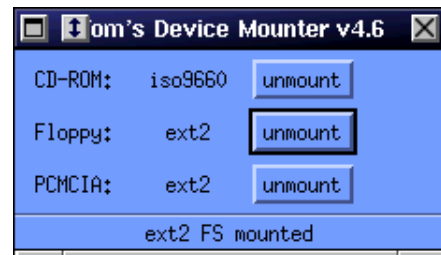


図 3 LessTif でコンパイルした `xmmount`

fstool[4][W³]

Tcl/Tk によるもので、なんと 1992 年作となつています。xvmount-1.0 が 1993 年 7 月なので、こっちの方が古いので驚きました。ただし、中で mount コマンドを呼び出しているのに過ぎません (xvmount 系は *mntent 関数を使って自前でマウントを行っており、正統といえるでしょう)。また、/etc/fstab の内容をそのまま表示してしまうので、却って不便かもしれません。



図 4 Tcl/Tk を用いた fstool

tkmount[5][W³]

この辺で、自作の虫がうずうずしてきました。マウント状態は mount コマンドを引数無しで実行して grep でデバイス名があるかどうかで判断し、mount/umount をトグルボタンと結びつけるといふ安易な設計 (いつもこんなばっかり) でいきましょう。もちろん、afterstep の wharf ボタンに収めるため 56×56 ピクセルの大きさに限定します。最小限の機能は Tcl/Tk で書くと 70 行弱で済みました。どうせエントリが 3 つなので、最初の方で devname(i) にマウントポイント名を埋め込んでしまいました。ここを自分の環境に合わせて書き換えて実行します (コンパイルの必要がないので、コンフィグレーションを書き換えるのと同じ時間で済みますからね)。ついでに xmounter をアテナウィジェットで書いてみましたが、どうしても 200 行位になってしまいます。Tcl/Tk はこういう場合にはやはり簡単でいいですね。



図 5 tkmount と xmounter : Wharf (56×56) ボタンに収まる大きさです

リスト 1. tkmount

```
#!/usr/bin/wish -f
#
# Tcl/Tk マウントツール
# /etc/fstab に従う user マウント状態のモニター
# Leave 時に sync, Enter 時に最新状態に update

set devname(1) cdrom
set devname(2) floppy
set devname(3) card

set sfont -adobe*-bold-r*-10*-p-*
set bgcolor "rgb:90/90/a0"
. configure -bg $bgcolor

proc check_mount {name flag} {
    upvar $flag mnt
    if {[catch {exec mount | grep $name}] == 1} {
        set mnt 0
    } else {
        set mnt 1
    }
}

proc tgl_mount {name flag} {
    upvar $flag mnt
    if { $mnt == 1 } {
        if { [catch {exec umount $name}] == 1 } {
            set mnt 0
        } else {
            set mnt 1
        }
    } else {
        if { [catch {exec mount $name}] == 1 } {
            set mnt 1
        } else {
            set mnt 0
        }
    }
}

set check_all {
    check_mount $devname(1) m1
    check_mount $devname(2) m2
    check_mount $devname(3) m3
}

wm geometry . 56x56
for {set i 1} {$i < 4} {incr i} {
    checkbutton .b$i
```

```

        -anchor w -width 56 -text $devname($i) \
        -font $sfont -highlightthickness 0 \
        -bg $bgcolor -variable m$i \
    place .b$i -x 0 -y [expr ($i-1)*18]
}
bind . <Enter>    { eval $check_all }
bind . <Leave>    { catch {exec sync} }
bind .b1 <Button-1> {tgl_mount /$devname(1) m1}
bind .b2 <Button-1> {tgl_mount /$devname(2) m2}
bind .b3 <Button-1> {tgl_mount /$devname(3) m3}
eval $check_all

```

wmmount[6] [W³](#)



図 6 WindowMaker 専用 wmmount : 右下の矢印ボタンで表示デバイスを切替えます . Floppy (左) と CD-ROM (右) の絵がなかなか Good です .

いままでは、デバイスの一覧が文字で表現されてました . wmmount は右下にある矢印のキーでデバイスを切替えて操作します . 中央右付近にあるデバイスの絵をダブルクリックすると外部コマンドが実行されるようになっていますから、ファイルブラウザ (FD) などを起動するように仕掛けておくといいかもしれません . WindowMaker 用なので大きさが 64×64 と少し大きく Wharf には使えません . マウントされているかどうかのチェックが独特で、stat 関数を使って変化を監視します . しかし、その判定がうまく機能しません . ソースを眺めて、439 行目に *strchr(mpstr, '/')=""\0"; という謎の 1 文を、素直に strcpy(mpstr, "/"; と書き換えてまあ動くようになりました . マウントの実行は mount/umount を呼び出しています .

asmount[7] [W³](#)

一つのデバイス (狙いは CDROM らしいのが絵柄で判ります) のマウントを実行するように設計されているのが asmount です . マウントされると、画面上のディスクが明るくなります . CDROM 以外のデバイスに対しは、引数に -m <mount point> を指定して使います . ただし、内部で mount コマンドを、「お試し mount」の (*) 形式で呼んでしまうため、ユーザー権限ではうまくありません (この形式は root しか実行で



図 7 After Step 用に設計された asmount : 一つのデバイスしか制御できません . マウントされるとディスクが明るくなります (右)

きません) . この不具合 (仕様 ?) は、該当箇所を一箇所ちよつと書き換えれば回避できます . オプション -s <sec> で更新の間隔を秒単位で指定してできるのは便利です .

参考文献

- [1] コンパイルの難しそうなソースの入手は Linux 用のパッチが整備されているので、Debian を利用しています .
<http://packages.debian.org/stable/utils/xvmount.html> [W³](#)
- [2] Public Domain のソースを探すなら ibiblio でしょう .
<http://www.ibiblio.org/linsearch/lsmis/xmount10.html> [W³](#)
- [3] <http://www.ibiblio.org/linsearch/lsmis/xmmount-4.8.html> [W³](#)
- [4] <http://www.ibiblio.org/linsearch/lsmis/fstool.html> [W³](#)
- [5] 筆者の FTP サイト
<ftp://ayapin.film.s.dendai.ac.jp/pub/X11/Mounts/> [W³](#)
- [6] <http://packages.debian.org/stable/utils/wmmount.html> [W³](#)
- [7] linux.org の検索結果
http://www.linux.org/apps/AppId_3168.html [W³](#)