



章を端末に表示して読み合うというものです。

### /dev/tty\* への書き込み

kterm などの端末エミュレータは文字の読み書きを /dev/tty0 などの端末デバイスファイルを通じて行います。また、現在使用している端末デバイスファイル名は、tty コマンドを引数なしで入力して得ることができます。X で kterm などを二つ開くと最初起動された方に /dev/tty0、次に起動された方に /dev/tty1 が割られるでしょう。そこで実験です。一方の端末上で他方の端末が使っているデバイスファイルに直接文字列を書き込んでみます。

```
echo Are you tty1? > /dev/tty1
echo 日本語もOK? > /dev/tty1
```

どうですか、tty1 を使っている kterm 上に文字列が表示されましたね。まあこんな調子で、他のユーザーの使用している端末に文字列を表示させることが原理的に可能なことは理解できました。しかし、デバイスファイル /dev/tty0 のパーミッションを調べると msg n の状態では

```
aya:~$ ls -l /dev/tty0
crwx----- 1 matuda users ...
```

となっていて、もちろん使用しているユーザー以外の読み書きはできないことになっています（できたら恐ろしい）。msg y の状態では group の書き込みパーミッションが立って crwx-w---- となり（通常）user グループメンバーからの書き込みのみ可能となります。

### write

そうゆうわけで、相手方が msg y の状態ならば、

```
write user@host ttyname
```

と、名前（ホストが別ならホスト名を@に続けて示します）、端末名（限定したいなら）を引数にして write を使うと、相手の端末に突然

```
Message from matu@aya.com on tty2 at 23:01 ...
```

などと割り込んで、一方的にメッセージを表示することができるのです。Ctrl-d や Ctrl-c を入力すると、相手方にも EOF が表示されて write は終了します。

### talk

write は一方的な書き込みでメッセージの交換とはいがたいコマンドでした。それに比べて、talk は誘いをかけてお互いの端末を上下に分割してそれぞれにメッセージを表示してくれるツールです。誘いをかける方のコマンドの引数は write と同じですが、誘いを受けた側にも選択の余地があるところが違います。talk と答えなければ会話が始まりません。誘った方は、相手が talk とコマンドを入力するまで待ち続けることになります。図には、kterm と rxvt での talk の様子を示します。kterm はその場で日本語変換できますからいいですが、rxvt では別の漢字変換窓が現れるのでちょっと使いづらいですね。talk を終了するには、終りの意志表示を伝えてから Ctrl-c します。



図 1 talk による端末上の会話

### okphone[2][W3]

rxvt で talk すると漢字入力が大変なだけでなく、入力もおかしくなることがありました（筆者の設定が悪いのが原因かもしれませんが）。ところで、会話では必ずしも漢字変換は必要ではなく、平仮名でも十分です（頭の中で漢字に置き換えることとなりますね）。okphone は平仮名変換内蔵の phone です。最初からローマ仮名

変換されるので、Ctrl-SPC などで漢字変換を起動してはいけません。また、マニュアルでは、読みやすくするために『わかち書き』をするよう推奨しています。表示に用いる漢字コードは起動時にも `-c[aejs]` と指定できますが、設定ファイル `~/ .phonerc` に

```
set code euc
```

と記述して、デフォルトを EUCJ にしておきましょう。英語版の作者は、Jonathan C. Broome (broome@ucb-vax.berkeley.edu) 氏で、かなパッチは、橘 (tachi@kaba.or.jp) 氏となっています。他にバグ取りで大きな貢献をした David J. Aladin 氏、Katsuya Odajima 氏の名前もマニュアルに挙がっています。

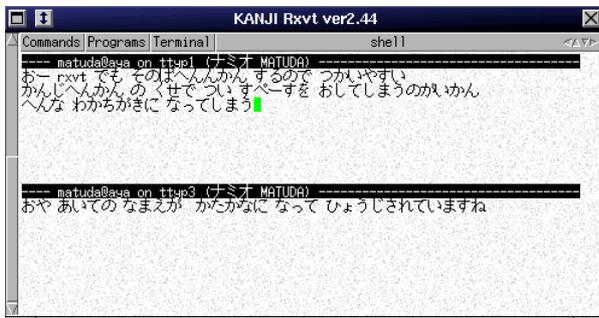


図 2 okphone : かな変換内蔵 phone

### xitalk[3] [W<sup>3</sup>](#)

仕事をしている時に、メッセージが端末画面に突然現れるのは嫌なものです。xitalk は、誘いがあった時に、別の端末画面を立ち上げる機構を提供してくれるツールです。仕組みは簡単に言うと、他の端末の mesg を禁止しておき、xitalk の所有する端末のみ mesg y として、外部からの要求を一手に引き受けるといふものです。普段は待機のアイコン (図上左) が表示されており、talk や write や phone からの接続要求があると、アイコンを変化させます (図上右)。左ボタンでクリックすると、細々としたダイアログが現れますが (図下)、あまり使うことはないでしょう。誰からの誘いかいちいち確認せずに、右ボタンでクリックして予め決めておいたアクション (kterm を起動して talk コマンドを発行するなど) を起す方が手短かだからです。

さらに、設定ファイル `~/ .xitalkrc` 内で AUTOANSWER を YES にしておくと、自動的にアクションが起動します。音声通知機能もありますし、これで talk や okphone を自由に楽しむことができるようになります。

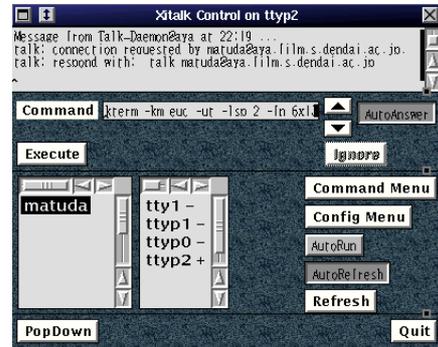


図 3 xitalk : talk や phone の中継ツール

すね。

### 居場所が判る xipmsg

ここまで紹介したツールは、予め相手がどのホストにいるか判っている場合に使うことができるものでした。相手がどのホストにいるか判らない場合には、who や rwho で所在を探さなければいけません。xipmsg はその手間を開放してくれます。xipmsg では、立ち上げ時に (あるいは更新コマンドで) ブロードキャストを使ってネットワーク内の全ての xipmsg ユーザーを検索します。そうして、得られた参加者一覧から相手を選んでメッセージを 1 つずつ送るツールです。メッセージを受けられるように、予め xipmsg を次のように起動しておきます。

```
xipmsg -name <name> -broadcast <bcast-address>
```

-name を省略すると現在のユーザー名がそのまま使われます。-broadcast を省略するとデフォルトの 255.255.255.255 が使われてしまいますから、相手がいると思われるサブネットワークのブロードキャストアドレスを指定して非難を受けないようにしましょう (もっとも 1 回だけなのであまり問題視されることはないと思います)。

### xipmsg[4] [W<sup>3</sup>](#)

使い方は簡単で、画面を見れば想像がつきます。まずメインウィンドウで名前リストから相手を選び送信ボタンを押します (左上図)。現れた送信ウィンドウ上でメッセージを入力し、必要ならばアイコンも (ボタ



図 4 xipmsg

ンを押すと候補が現れます) 変更して、送信ボタンを押しますと(右上図)相手側にメッセージが表示される(左下図)というものです。返事を書くならば、返信ボタンを押して送信ウィンドウをポップアップさせ、以下繰り返すです。

なお、Xipmsg は IP Messenger [5] [W<sup>3</sup>](#) の X11 への移植版です。元々 MS-Win で開発され Mac にも移植されていますので、これらのユーザーとも通信できます。

## 音声で

さてマルチメディア化が進んだ現在では、ネットワーク越しに音声で会話するためのツールがあってしかるべきですし、商品化されたものはあります。しかし Linux ではフリーなもの意外と少ないようです。

## mtalkf [6] [W<sup>3</sup>](#)

音声を伝達するには、ネットワークが高速でなければいけません(画像はもっと深刻ですが)。もちろん、途切れないように、逐次圧縮して送り出す工夫も必要です。ところで、一端録音した音声データ全体を圧縮して送り出す仕様にするると随分プログラムが簡単になります。mtalkf はそのようなシンプルな設計となっています。talk というよりは一方通行の音声版 write です。mtalkf ではデーモン mtalkfd が inetd から呼び出されます。したがってあるホストで matlkf からの接続を受けられるようにするには /etc/inetd.conf に次のように登録し、

```
mtalkfd stream tcp nowait root
/usr/sbin/mtalkfd mtalkfd
```

さらに /etc/services にポート番号 5105 を

```
mtalkfd 5105/tcp # full-rate mtalk
```

と登録して、inetd を

```
kill -HUP <inetd の pid>
```

して、初期化しなければなりません。

これで相手が mtalkf の要求を受けられる状態になりましたから、こちらから mtalkf を

```
mtalkf 相手の hostname
```

と起動し、一方的に音声を流します。録音と送信の切替は Return キーを使います。以下に、実行例を示します。

```
aya:~$ mtalkf aya
Listening: Hit [RETURN] to start your message
                                for aya
(RETURN キーを押して録音開始)
mtalk: starting record
Recording: Hit [RETURN] to stop
(最後まで録音したら RETURN キーを押す)
mtalk: ending record
mtalk: connecting to aya
mtalk: sending data
>001 cb server ready
>003 Reading 10240 bytes
Listening: Hit [RETURN] to start your ...
```

会話をしたいなら、受信側も mtalkf を起動して音声を送り付けます。普通のサウンドカードでは録音と再生が同時にはできないので、トランシーブで交信する時のように『... マシタッ・ドウゾッ』と合図を送って、喋るタイミングを計る必要があります。

## vat [7] [W<sup>3</sup>](#)

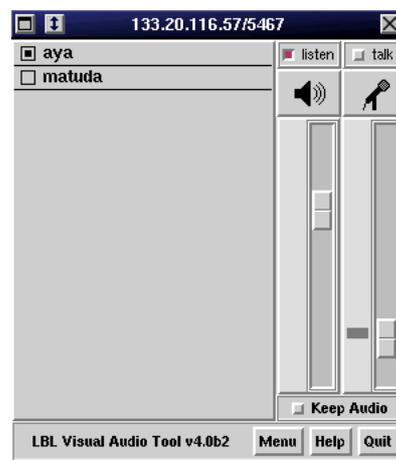


図 5 vat : 音声会議ツール

vat はネットワーク接続したホストで会議を行うためのツールです。ローレンスパークレ

国立研究所ネットワーク開発部のプロジェクト成果 (<http://www-nrg.ee.lbl.gov/vat/>) で、Tcl/Tk ベースの使い易い GUI を持っています。元来、会議用なので非常に多くに機能が備わっていますが、ここでは二人でおしゃべりするための使い方のみ説明します。まず、接続先とポート番号 (5000 番以降がよいでしょう) を引数にして

```
vat hostname/5467
```

のように起動します。すると、左側には会議参加者の一覧が、右側にスピーカ (listen) とマイク (talk) のトグルスイッチやスライドボリュームが表示されます。しかし全二重でない普通のサウンドカードでは、一般に listen (再生) と talk (録音) を同時には使えません。どちらか一方の機能に切替えて使わなくてはならないのです。ところが vat ではトグルスイッチで一々切り替えなくても良い方法があります。トグルスイッチを図のように、listen を ON とし talk を OFF とします (デフォルト状態)。これは再生のみ可能な状態ですが、右マウスを押している間は talk が ON となり話すことができます。この時 listen のスイッチは ON のままですが実際には切り離されます。

コンパイルはちょっとだけ工夫が要りました。日本語化 Tcl/Tk のライブラリ名\*jp.so では configure できないのです。次のように、英語版へのリンクを作成して回避しなければなりませんでした。

```
ln -fs libtcl8.0jp.so libtcl8.0.so
ln -fs libtk8.0jp.so libtk8.0.so
```

なお、Tcl/Tk のバージョンに関しては、筆者の環境ではたまたま 8.0 が導入されていましたが、tcl7.6/tk4.2 以上で使用可です。

## 次回は

文字、音声ときたらつぎは映像を期待してらっしゃる方もおられるでしょうが、Linux では使用可能な映像の入力デバイスの種類が少なく、筆者も qcam くらいしか経験がないのでどうにもお話しになりません。いずれまたの機会にということまでひとつ宜しく...

さて次回ですが、デスクトップ小物ネタもそろそろ底を尽いてきました。小物とは違うのですが、あまりに自然で、しかしながらそれ無しには仕事ができない空気みたいな存在、文字端末エミュレータについて、話しを盛り上げてみようと思います (盛り上がるでしょうか、ちょっと心配 (^;))。

## 参考文献

- [1] 安岡さんのホームページ  
<http://kanji.zinbun.kyoto-u.ac.jp/~yasuoka/program.html> 
- [2] 小田島さんの okphone ページ  
<http://mlnews.com/okphone/>   
上記から入手したセットだけではインストールできない場合の Linux 向けのパッチ 
- [3] XIntercepTalk の公式サイト  
<http://www.csn.ul.ie/~caolan/docs/xitalk.html> 
- [4] 開発者 (神田さん) の XIP Messenger ページ  
<http://hp.vector.co.jp/authors/VA004959/xipmsg/> 
- [5] Xipmsg はメッセージ送受信サービス IP Messenger の X11 版です。IP Messenger の本家  
<http://www.asahi-net.or.jp/~VZ4H-SRUZ/ipmsg.html> 
- [6] ibiblio の FTP アーカイブ  
<http://ibiblio.org/pub/linux/apps/sound/talk/> 
- [7] バークレイ国立研究所の Vat 公式サイト  
<http://www-nrg.ee.lbl.gov/vat/> 