
インターフェイスの街角 (19)

PocketBSD

増井俊之

PocketBSD とは

PocketBSD とは、日本電気が販売している携帯端末「Mobile Gear」(図1)で動く FreeBSD システムです。現在、Mobile Gear という製品名で 2 系統の製品が販売されています¹。外見や内蔵ソフトは似ていますが、2 年ほど前に発売された初代の Mobile Gear(MC-MK12 や MC-MK32 など)は 486 CPU の DOS に PDA 的な皮をかぶせた製品、昨年から今年にかけて発売された新版の Mobile Gear(MC-R500 や MC-R700 など)は MIPS CPU 上に Windows CE を載せた製品で、中身のハードウェアやソフトウェアはかなり違います。初代の Mobile Gear は、軽くて手頃な DOS マシンとして一部で人気が高いようですが²、たけむらさんという方がこれに FreeBSD を載せたものが PocketBSD です³。Mobile Gear にはハードディスクはありませんし、メモリも MK12 は 4MB、MK32 は 8MB です。しかし、PC カードのフラッシュメモリをディスクとして使い、FreeBSD を動かしています。

PocketBSD は、電池で動く UNIX という点で魅力的ですが、FreeBSD 2.2.5 の動いているノート PC を経由して PC カードにシステムを構築しなければならないなど、インストールに手間がかかる(ような印象があった)ため、導入をためらっていました。しかし、『いつでもどこでも UNIX! PocketBSD インストールキット』[1](以

図1 Mobile Gear MC-MK12



下、「PB インストール・キット」)が出版され、手軽にインストールできるようになりました。

この本に従って PocketBSD を Mobile Gear にインストールし、1 カ月ほど使用したところですが、たいへん快適に携帯 UNIX が使えることに満足しています。ディスク容量が足りなかったり、X ウィンドウや PC カードが使えないといった制約はありますが、ほぼ普通の UNIX として利用できます。

インストール

インストール方法は PB インストール・キットに詳しく書かれていますが、インストーラがよくできているためごく簡単です。DOS フォーマットのフラッシュメモリ・カード上に PocketBSD のインストーラをコピーし、それを Mobile Gear の外部ソフトとして起動すると、インストーラはメモリカード上にファイルシステムを新たに作り、必要なファイルをすべて生成します。フラッシュメモリ・カードにインストーラをコピーできれば、あとは Mobile Gear 本体だけあればいいので、手許に Windows 95 のノート PC しかなくても PocketBSD をインストールできます。

1 <http://www.pc98.nec.co.jp/Product/mg/lineup.htm>

2 「とことん使おう Mobile Gear」(<http://www02.so-net.ne.jp/~suzuden/>) や 「時計じかけ別館 MobileGear」(<http://www.asahi-net.or.jp/~bx6y-asi/aki/mg/index.htm>) などに詳しい情報があります。

3 <http://www02.u-page.so-net.ne.jp/ca2/takemura/>

インストーラでは最小限のファイルしか Mobile Gear にコピーされないため、PocketBSD を使うときに必要な Mule や PPP などは pkg-add で追加インストールしなければなりません。フラッシュメモリ・カードの一部を DOS 領域として残しておき、その領域を使ってノート PC の CD-ROM からパッケージを PocketBSD に移してこれらをインストールします。

おもな用途

文書の作成

PocketBSD でもっとも嬉しいことの 1 つは、デスクトップの UNIX マシンと同じ環境で文章やメールが書ける点です。ふだん、私は Mule と SKK を利用していますが、それ以外の環境では文章をあまり速く書けません。Windows 95 のノート PC なら Mule for Windows がありますが、Windows CE マシンや PDA では Mule が使えないのが問題でした。Mobile Gear には、もともと「MG エディタ」というエディタが付属しています。さいわい、MG エディタではキー割当てをカスタマイズできるので、カーソルの右移動に Ctrl-F を割り当てるなどしてなんとか Mule ふうに使っていましたが、文章のカット & ペーストや日本語入力的方式などはまったく異なりますし、複雑な Emacs コマンドはもちろん使えません。そんなわけで、私の場合は Mule を使うときとくらべてどうしても効率が悪くなっていました。しかし、PocketBSD の Mule を利用すれば、画面サイズ以外は完全にデスクトップと同じ環境で文章が書けるので、外出中でも、寝ころがりながらも、快適に書けるようになりました。

PocketBSD では TeX も利用できます。DVI ファイルを PocketBSD 上で閲覧するツールはまだ揃っていませんが、dvi2tty を使っておおよその結果を見ることはできます。Mobile Gear 用の dviout⁴を使うか、Ghostscript の PocketBSD 用ドライバを作成すれば、プレビューもうまくできるようになるでしょう。

PocketBSD では、普通の FreeBSD と同様に Canna や SKK のような日本語入力システムが使えます。通常の DOS/V マシンに FreeBSD をインストールした場合な

どと異なり、PocketBSD ではコンソールに直接日本語を表示できるので扱いも簡単です。

プログラム開発

PocketBSD では、GCC や Perl などを用いてプログラム開発ができるのも嬉しいところです。電車のなかや公園などでプログラムをデバッグしたり、思いついたアルゴリズムを実験したりできるのは気持ちがよいものです⁵。

ちょっとしたシステムのプロトタイプを作成する場合など、端末シーケンスや curses で作るようにしておけば、デモ用マシンとしても使えます。おかげで、UNIX 端末で動くプログラムのデモが気軽にできるようになりました。

通信

外部とデータをやりとりするために、Mobile Gear には PC カードスロット、モデムポート、シリアルポート、赤外線ポートがあります。残念ながら、PocketBSD では PC カードスロットのフラッシュメモリが外せず、赤外線ポートには現時点では対応していません。通常の通信には、シリアルポートか内蔵モデムを利用します。

モデムやシリアルケーブル経由での接続には、以前は tip や cu、kermit などのコマンドがよく使われていましたが、最近は PPP による IP 接続が一般的なようです。PocketBSD では、IJ-PPP パッケージを用いてプロバイダや別の UNIX システムに比較的簡単に接続できます。

Mobile Gear の内蔵モデムは最大 9,600bps で、PPP 接続はあまり快適とはいえません。しかし、シリアルケーブル経由で別の UNIX システム、あるいは MN-128 SOHO や RT50i などのダイヤルアップ・ルータに PPP 接続する場合は 38,400bps まで使えるため、通信もかなり快適になります。ただし、数 MB 以上のデータを転送する場合は、シリアル経由よりも PC カードで受渡しをするほうが現実的でしょう。

メールと Web

PPP で接続すれば、電子メールや WWW が利用できるようになります。従来のデスクトップ UNIX マシンの

4 <http://www.t3.rim.or.jp/~patakuti/mg/dviout/>

5 昨年の夏、Windows PC で「海辺でプログラミング」に挑戦したのですが、明るい日光の下では画面が見えませんが、電池の残量も気になって、とても快適とはいえませんでした。その点、PocketBSD なら大丈夫そうな気がします。

図 2 NetFront



多くは常時 LAN に接続されており、メールの返事を書いているあいだもネットワークにつながっているのが普通でしたから、メールの送受信時だけネットワークに接続するといった「貧乏な使い方」はあまり想定されていなかったようです。モバイル環境で UNIX を使うときは、PDA や Windows PC などと同じようにメールの送受信時だけ接続し、返事を書くあいだはオフラインにしておきたいものです。

PB インストール・キットでは、オフラインでのメールの読み書きに popclient と gnsPOOL の利用を推奨しています。popclient はメールサーバーに POP で接続して未読メールを読み出し、指定したファイルに格納します。その後、接続を切って MH の show や comp などでメールの読み書きをおこない、今度は gnsPOOL を使って SMTP でメールを送るという手順になります。この手法の場合、接続先を複数指定できないといった制約はありますが、とりあえず使いたいときには設定が簡単で便利です。

PB インストール・キットにはアクセス⁶の開発した Web ブラウザ「NetFront」の試用版が付属しています(図 2)。NetFront はサイズが小さいわりには高性能なブラウザで、各種の携帯端末で採用されています。

NetFront よりも手軽な Web ブラウザとしては、テキストベースの Lynx や伊藤章則さんの開発した w3m⁷などがあります。

携帯端末としての使用

前回は UNIX 上での個人情報管理の手法を紹介しましたが、同じ方法を PocketBSD に適用すれば PDA のような使い方ができます。

前回紹介したような方法でスケジュールなどを HTML 文書にしておけば、NetFront や w3m などの Web ブラ

ウザで閲覧することもできます。メモや住所録などのデータは、通常の UNIX ファイルとしてデスクトップの場合と同様に管理しておけばいいでしょう。

携帯端末では、辞書機能が必要とされることも多いようです。Mobile Gear は英和辞書や国語辞書を内蔵 ROM 内にもっているため、これらを PocketBSD でも使えば便利です。残念ながら、PB インストール・キットで配布されているバージョンの PocketBSD では UNIX プログラムから内蔵 ROM の辞書に直接アクセスすることはできません。しかし、新しいバージョンの PocketBSD ではこれができるようになる可能性もあります。現状の PocketBSD でも、ROM 内の辞書をいったん DOS 上で読み出して PocketBSD のファイルとしてコピーしておけば、(そのぶんフラッシュメモリを消費しますが) Mobile Gear の辞書を PocketBSD でも利用できるようになります。このためのツールとして、mgedic⁸、dictmg⁹などが開発されています。

ほかのシステムとの同期

PDA とデスクトップ PC を併用している場合、データの整合性を保つ方法が大きな問題になります。PalmPilot や Windows CE などでは、そのための各種のハードウェアやソフトウェアが作られています。しかし、デスクトップ・マシン上のシステムの数×PDA の種類だけ同期/データ変換システムを用意しなければならないのは大変です。事実、PDA 関連のメーリングリストでは、「PDA のソフトウェアとを同期させるにはどうすればいいか?」といった質問が飛びかっているようです。

一方、デスクトップ・マシンでも PDA でも UNIX を使っていれば、データの整合性を保つのも簡単です。データの変換は不要なので、ファイルシステムを比較し、両者が同じになるようにするだけです。

複数の UNIX システム間でファイルの整合性をとるには、Syncdir¹⁰や rsync¹¹など、各種のツールが利用できます。rsync は rcp を拡張したもので、ローカルマシン

6 <http://www.access.co.jp/>

7 <http://ei5nazha.yz.yamagata-u.ac.jp/>

8 <http://www.asahi-net.or.jp/~km5n-tkhs/>

9 <http://www.ss.ij4u.or.jp/~you/pbsd/intro/note09-dictmg.html>

10 <http://www.etl.go.jp/~tatebe/software/syncdir/>

11 <http://samba.anu.edu.au/rsync/>

のファイルとリモートマシンのファイルとを比較し、新しく生成されたものだけをコピーするという指定が可能です。たとえば、PocketBSD の Memo というディレクトリに下記のような Makefile を置いておき、適宜 make を起動すれば、リモートマシン “bird” の同名のディレクトリの内容と整合性を保つことができます。

```
# rsyncでリモートマシンとディレクトリの同期をとる
#
# -a : (アーカイブ)ファイルの属性を揃える
# -v : verbose
# -u : 新しいもののみコピー
# -z : 転送に圧縮を使うか?

RSYNC=/usr/local/bin/rsync
REMOTEDIR=bird:/user/masui/DOC/Memo/
EXCLUDE = --exclude RCS \
          --exclude '*~' \
          --exclude '\#*'
sync: put get
put:
    $(RSYNC) -avuz $(EXCLUDE) . $(REMOTEDIR)
get:
    $(RSYNC) -avuz $(EXCLUDE) $(REMOTEDIR) .
```

グラフィックスとウィンドウ

PocketBSD では、mmap() でコンソールをメモリにマップして画面を直接読み書きすることができます。たとえば、下記のようなプログラム¹²を作れば、PGM形式で画面のスクリーンダンプが撮れます(誌面の都合上、⇒で折り返しています)

```
#include <stdio.h>
#include <sys/mman.h>

const unsigned char *intensity[4] =
    {"240 ", "180 ", "120 ", "0  "};
char buf[100];

main()
{
    char *mem;
    int i,j,b,y,val;

    mem = mmap(0,0x10000,PROT_READ, ⇒
              MAP_PRIVATE,0,0);
```

12 PocketBSD 開発者のたけむらさんがメーリングリストに投稿した記事 (pbsd-dev:01721) (<ftp://bronze.is.titech.ac.jp/pub/PocketBSD/ml/pbsd-dev/pbsd-dev.01700.tgz>) をもとにしています。

図 3 8ドットフォントを使用した mgterm



```
if(mem == MAP_FAILED) exit(1);

printf("P2\n640 240\n255\n"); // PGM header
for(y=0;y<240;y++){
    for(i=0;i<40;i++){
        for(j=0;j<4;j++){
            for(b=0;b<4;b++){
                val = (mem[y*0x100+i*4+j]>>(b*2)) ⇒
                    &0x03;
                strcpy(buf+(j*4+b)*4,intensity[val]);
            }
        }
        strcpy(buf+64,"\n");
        fwrite(buf,1,65,stdout);
    }
}
}
```

鈴木康司さんは、この機能を用いて PocketBSD 画面の上にグラフィックスを表示するためのライブラリ「MGL」を開発しています¹³。MGLは、直線や矩形などを描画する出力関数とキー入力を取得する入力関数から構成されています。ポインティング・デバイスは利用できないものの、MGLを使えば PocketBSD 上で比較的容易にグラフィカルなアプリケーションが作成できます。

MGLの配布パッケージには、やはり鈴木さんが開発した MGL を利用した端末エミュレータ「mgterm」も含まれています。mgterm は、Linux のコンソールで漢字を表示するために使われている VT100 エミュレータ「kon」をもとに PocketBSD+MGL 用に開発された端末エミュレータです。8ドットから 24ドットまでのいろいろな大きさのフォントが表示できます。8ドットフォントであれば、Mobile Gear の小さな画面でもかなりの量の文字を表示することができます(図 3)

「かたむきモバ」のページを運営している古川 陽さんは、MGLをもとにしたウィンドウ・ライブラリ「MNW」を配布しています¹⁴。MNW は PocketBSD 上でイベント

13 <http://www.kt.rim.or.jp/~suz/pbsd/>

14 <http://www.ss.ij4u.or.jp/~you/pbsd/task/mnwapi.html>

図 4 スケジュール管理ソフト「タスク」



駆動スタイルのウィンドウ・プログラミングを支援するためのライブラリで、立体表示文字、ボタン、編集枠などのツールが用意されています。古川さんはこのほかにも Mobile Gear の内蔵辞書を検索する「辞書」やスケジュール管理用の「タスク」(図 4)など、MNW を用いた多数のグラフィカル・アプリケーションを公開しています。

佐野さんの開発した X ウィンドウ・システム上で動く MGL 互換ライブラリ¹⁵を利用すれば、MGL を用いた PocketBSD プログラムを普通のデスクトップ・マシンで開発できるようになります。アーキテクチャが異なる携帯マシンとデスクトップ・マシンで同じプログラムを動かすのには苦労しますが¹⁶、UNIX で MGL を使うかぎり、そのような心配はなくなります。

その他のユーティリティ

図形エディタ

Mobile Gear にも図形エディタが付属していますが、PocketBSD 上でも絵を描けるようにしようという試みがあります。「機動歯車」のページ¹⁷を運営している中馬さんは、MGL と MNW を利用した PocketBSD 上のビットマップ・エディタ「うどん」を公開しています¹⁸。ポインティング・デバイスが利用できないマシンで図形エディタが実用になるかは難しいところです。数年前、筑波大学の久野 靖さんがキーボードだけで編集操作ができる X 上の図形エディタ「k2d」¹⁹を提唱していましたが、このような手法を使えば PocketBSD でも無理なく絵が描けるようになるかもしれません。

15 <http://www.yk.rim.or.jp/~yukihiko/pbsd/>

16 Windows と UNIX で同じプログラムを利用するには、OpenGL/GLUT、あるいは Java を使ったりと、いろいろな工夫が必要で困ります。

17 <http://ha3.seikyoku.ne.jp/home/Chum/>

18 <http://ha3.seikyoku.ne.jp/home/Chum/dev/myworks.html>

19 キーボードだけを使う kdraw という意味です。

図 5 MGTreemap

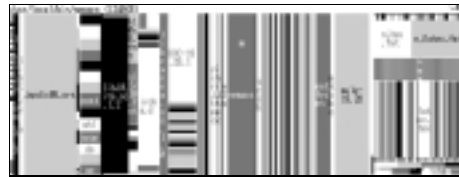


図 6 mkobo(MobileGear 版 xkobo)



ディスク使用量の視覚化

PocketBSD では、PC カードのフラッシュメモリをディスクとして使うため、ディスクの無駄遣いはしたくありません。そこで、MGL を用いてディスクの使用状況を表示するツール「MGTreemap」を作ってみました²⁰。

このプログラムで使っている Treemap [2] は、メリーランド大学にいた Brian Johnson 氏が開発した情報視覚化手法で、大きな階層情報を小さな矩形領域内に表示するものです。ファイルやディレクトリの大きさに応じて矩形を再帰的に縦横に分割していき、図 5 のように、各ファイルの占める大きさが分かるようになっています。図 5 は /usr/local 以下のファイルの大きさを mgtreemap コマンドで表示したところですが、/usr/local/bin がもっとも大きな領域を占めており、なかでも mule や nemacs が大きな場所をとっていることが一目で分かります。

ゲーム

PocketBSD では、nethack のような端末ベースのゲームはもちろん、四川省や xkobo(図 6)などのグラフィカルなゲームも移植されています²¹。

注意点など

PocketBSD にはかなり満足していますが、問題がないわけではありません。以下、思いつくままにいくつか挙げてみます。

20 <http://www.csl.sony.co.jp/person/masui/PocketBSD/mgtreemap.html>

21 <http://www.asahi-net.or.jp/~km5n-tkhs/>

- 通信が遅い

本体の処理速度にはそれほど問題を感じませんが、Webページの閲覧やメールの送受信の通信はほかのPDAにくらべてもかなり遅く感じます。

- 電池の持続時間

おそらく省電力の工夫がされていないためか、DOS版のMobile Gearにくらべて電池がかなり速く減っていきます。通常の単3電池が使えるので気分的には安心ですが、携帯端末なのに電池の残量を気にしなければならないのはちょっと残念です。いまのところはニッケル水素電池を本体でちょくちょく充電しながら利用していますが、長時間、屋外で使うようなときは電池を別途用意しなければならないかもしれません。

- レジャー時のハングアップ

スリープ状態からレジャーするとき、フラッシュメモリの立ち上がりが遅いせい、ときどきキーを受け付けなくなることがあります。電源キーを押してレジャーしたあと、時間を置いて操作すれば大丈夫かもしれません。

- PPPの相性

PocketBSDをほかのマシンにPPP接続しようとしたところ、シリアルによる通信がオーバーフローしてうまくいかない場合があります。

- 立って使うときのキー入力

Mobile GearのキーボードはPDAとしては大きいので、机などの上に置いて使う場合はいいとしても、電車のなかなどで立っているときにはややおっくうです。とくにMuleなどを利用する場合、立ったままCtrlキーと別のキーを押したりするのはかなり困難です。片手でも使えるような操作体系を考える必要があるかもしれません。

おわりに

まだ新しいシステムのせい、上記のような問題は残っていますが、UNIXがPDAで使えるPocketBSDはたいへん魅力的です。現在、PDAのOSの主導権を握るために各社が熾烈な競争を繰りひろげています。私としては、ぜひともUNIXにPDA用OSの本命になってもらいたいと思っています。

今回の記事は、ほとんどPocketBSD上のMuleで書きました。キーボードと画面が小さいぶん、書く速度はデスクトップ・マシンの場合とくらべてやや遅くなりますが、どこにいても慣れた環境で文章を書けるのは嬉しいかぎりです。

PocketBSDを利用し始めるまで、本当に使いものになるのかなとか、軽いノートPCのほうがいいのではと思っていましたが、実際に手にしてみると電池や振動を気にしながらノートPCを持ち歩くのにくらべると圧倒的に気楽です。機能の面ではギガバイト級のハードディスクと大きなカラー液晶画面をもつノートPCのほうがはるかに有利ですし、大きさも極端に違うわけではありません。しかし、ディスクを内蔵せず単3電池で動く点は精神的にも楽な気がします。ノートPCを自転車のカゴに放り込んで移動するのは度胸が要りますが、PocketBSDならば大丈夫そうです。現在のノートPCの多くは、明るい場所でカラー液晶画面が見にくいのですが、その点、Mobile Gearの反射型液晶なら大丈夫です。野山に出て、思うぞんぶん文章を書いたりプログラムを作ったりすることもできるでしょう。私の場合、動画を見たりプレゼンテーション資料を作ったりグラフィカルなプログラムを開発する機会を除けば、ほとんどの仕事はPocketBSDで十分なので、短期間の出張や外出には手放せないものになりそうです。ポケットボード並みに流行してほしいものです。

日本電気はDOS版のMobile Gearの生産を続けるそうですし、Windows CE版のMobile GearにUNIXを移植しようとしている人もいるようです。携帯端末のOSとしてのUNIXの今後に期待したいところです。

(ますい・としゆき ソニー CSL)

[参考文献]

- [1] 榎 隆『いつでもどこでもUNIX! PocketBSD インストールキット』、アスキー、1999年
- [2] Brian Johnson and Ben Shneiderman, Treemaps: A space-filling approach to the visualization of hierarchical information structures, *Proceedings of the 2nd International IEEE Visualization Conference*, pp.284-291, San Diego, October 1991