視覚障害者の PC–UNIX の導入に関する考察

平成 14 年 2 月 5 日

情報電子工学科
唐澤 洋介

目 次

1 はじめに **1**

2 視覚障害者のコンピュータ利用環境 **2**
2.1 視覚障害者のための OS 3
2.2 マニュアルに関する問題 5

3 視覚障害者の OS インストール環境 **7**
3.1 インストールをするための質問事項 10

4 視覚障害者の PC–UNIX 使用環境 **12**
4.1 PC–UNIX における日本語環境 12
4.2 視覚障害者のための日本語環境 16
4.2.1 シェル 17
4.3 シェルの機能 18
4.3.1 視覚障害者の使いやすいシェル 19
4.4 視覚障害者に使いやすい PC–UNIX 環境 20

5 おわりに **21**

参考文献 **23**

概要

視覚障害者がコンピュータを利用することは基本的には困難である。それでも、職場や学校のみならず個人でもコンピュータを所有し利用しているユーザは多く存在する。しかし、現在はどのような利用環境でも GUI ベースの OS である MS–Windows が主流であり、視覚障害者にとっては必ずしも利用しやすい環境とはいえない。そこで、GUI ベースの OS と比較して使いやすいと考えられる CUI ベースの OS として PC–UNIX を取り上げる。PC–UNIX の導入に関して、視覚障害者のユーザ 1 人での OS のインストールが困難なこと、音声利用による日本語環境、コンピュータ利用においてほとんどのユーザが必要とするマニュアルが視覚障害者には利用が不便であることを問題として取り上げ、それについての現状と解決方法を考察した。また、PC–UNIX において、シェルはユーザインターフェイスであり、シェルの使用によりすべての作業が行なわれるといえる。しかし、視覚障害者が現在の対話型環境のシェルを利用するには、音声化の問題と音声化しても使いやすくならないという問題がある。そこで、その解決方法の一例としてシェルの日本語化を提案し、使いやすい対話型シェルにするための考察をした。

1　はじめに

パソコンのユーザインターフェイスは大きくグラフィカルユーザインターフェイス (GUI) と キャラクタインターフェイス (CUI) の 2 つに分けることができる。

GUI を用いた OS として MS–Windows と Mac OS がありアイコンでソフトウェアやファイルを管理し、マウスの操作を主としている。また、グラフや図形の描写に優れていて、画面情報を見ることのできるユーザにとっては非常に使い勝手の良い OS である。

一方、CUI を用いた OS として MS–DOS と PC–UNIX があり基本的にコマンドを入力することによって様々な作業を行なう。PC–UNIX では、 X ウインドウシステムを使えば GUI 環境でパソコンを利用することも可能である。これらの OS では、ある程度のコンピュータ知識及びコマンドの理解が必要であり、容易に扱うことはできない。

視覚障害者がコンピュータを利用することは基本的に困難であるが、職場や学校のみならず個人でもコンピュータを所有し利用しているユーザは多く存在する。現在はどのような利用環境でも GUI ベースの OS である MS–Windows が主流であり、視覚障害者にとっては必ずしも利用しやすい環境とは言えない。

また、視覚障害者のコンピュータ利用において音声は重要な役割を持つ。音声出力は、文章のみを読んでくれる音声読み上げソフトと、文章のみならず GUI 画面の情報をある程度読んでくれるスクリーンリーダーソフト (画面情報音声化ソフト) の 2 つに大きく分けられる。現在、スクリーンリーダーソフトは必ずしも使いやすいものとは言えない。また、PC–UNIX の場合、音声出力にはコンピュータが 2 台必要である。

このように、現在のパソコン環境は OS 面からも音声化の面からも視覚障害者のユーザにとって利用が容易な環境とはいえない。

視覚障害者のコンピュータ利用にとって困難なことは以上だけではない。マニュアルの利用、OS のインストールもまた困難なものである。

マニュアルに関する問題として、コンピュータ利用にはマニュアルは必要であるが、視覚障害者がマニュアルを使用するには点訳や音声を利用しなければならない。しかし、現在のところボランティアに頼ることが多く、点字や音声に直したマニュアルを入手するには時間がかかる。また、マニュアルのグラフィカルな部分は理解できないため、画像などを多用しているマニュアルは視覚障害者には向いていない。

また、OS のインストールの問題として、視覚障害者の GUI インストーラ利用の不便さがある。GUI インストーラは、主にマウスを使うため視覚障害者には利用が容易ではない。最近では、PC–UNIX の中でも Linux 系は GUI のインストーラが実装されるようになり、視覚障害者のユーザにはますますインストール環境は不向きなものになっている。そのような中、PC–UNIX の BSD 系はいまでも CUI ベースのインストーラであり、視覚障害者がインストールすることは可能な環境であるといえる。

以上のようなことから、視覚障害者のユーザには GUI ベースの OS と比較して CUI ベースの OS のほうが利用しやすいと考えられる。そこで、PC–UNIX を取り上げ、視覚障害者にとってよりよいコンピュータ利用環境について考察する。

2　視覚障害者のコンピュータ利用環境

視覚障害者は、弱視者と全盲者の 2 者に大別できる。

- 弱視者

  弱視者のユーザは、文字の大きさやコントラストや配色をカスタマイズすれば、コンピュータを利用することが可能である。カスタマイズのみで晴眼者同様にコンピュータを利用できる弱視者もいるが、様々な視覚障害を総合して弱視としているので、すべての弱視者がカスタマイズすればコンピュータを利用できるという訳ではない。

- 全盲者

  全盲者のユーザは、音声や点字を用いなければコンピュータを利用できない。そのため、音声化ソフトあるいは点字ディスプレイが必要である。

  全盲のユーザの中には、NEC の PC98 シリーズのコンピュータを使用しているユーザが多い。これは、音声出力を使えるフリーソフトが 他の AT 互換機より少し多いためだそうである。

以下に視覚障害者のコンピュータ利用環境をオペレーティングシステム (OS) の分類によって考察する。

- MS–Windows 環境

  基本的に、視覚障害者のユーザが、マウスを操作し画面情報にアクセスすることは困難である。画面情報を目で見て確認できない視覚障害者のユーザのためにスクリーンリーダーソフト (画面情報音声化ソフト) があるが、最近のスクリーンリーダーソフトは、マウスカーソルをマウスの代わりにテンキーで動かすことも可能である。また、グラフなどの視覚障害者にとって理解しにくい画面情報も特殊なソフトウェアや点字ディスプレイを使えば知ることができるようである。さらに、現在のスクリーンリーダーソフトでは、アイコンを音声で表すこともできるようになっている。しかし、新しくアイコンを作った場合にはユーザ自らが登録しなければならない。

  スクリーンリーダーソフト対応のアプリケーションソフトも増えてきて視覚障害者のための環境は整いつつあるといえる。

  また、同じ GUI 環境の OS として Mac OS がある。Mac OS にはテキストの音声読み上げソフトはあるがスクリーンリーダーソフトはない [7] ので、全盲者のユーザには利用が難しい。弱視者のユーザに対しては、キーボードやマウスを使いやすくするイージーアクセスという拡張機能や画面を拡大させるクローズビューという拡大機能がある [8] ので利用できるかもしれない。

- MS–DOS 環境

  キャラクタベースの OS であるため、視覚障害者のユーザに向いているとされている。また、音声化ソフトや音声対応のソフトウェアにもまだまだ使えるものが多く

あるため、視覚障害者のユーザの中には、未だに MS–DOS を使用しているユーザも少なくない。しかし、MS–DOS は、すでに新しいソフトウェアは開発されておらず、扱っている販売店も少なくなってきていて入手は困難になりつつある。また、MS–DOS に対応していない周辺機器も増えてきている。

- PC–UNIX 環境

  視覚障害者のユーザが PC–UNIX を使用する場合、現在、PC–UNIX には使いやすい音声化システムがないため、PC–UNIX がインストールされているパソコンとスクリーンリーダーソフトがインストールされている他のパソコンとを接続し、そのパソコンから PC–UNIX の画面情報を音声として出力させている。この点が他の OS と比較して不便だが、CUI ベースでフリーの PC–UNIX もあり、また視覚障害者に有用なフリーソフトウェアも充実しているため、PC–UNIX を使用している視覚障害者のユーザも少なくない。また、 MS–DOS を利用しているユーザは、同じテキストベースの OS なので PC–UNIX への移行も比較的容易だと思われる。

  また、Linux には Blinux という視覚障害を持つ Linux ユーザをサポートするプロジェクトがあり [9] メーリングリストも活発だが、現在のところ日本語環境はなく英語のみである。また、Blinux とは別に日本語で PC–UNIX 環境を構築できるものとして L–Voice[12] という視覚障害者のためのコンピュータ利用支援システムがあり、特徴は以下の通りである。

  1. ディスプレイ上の文字列を音声で読み上げ、視覚障害者のコンピュータ利用を可能にする
  2. アプリケーションに依存しない音声読み上げ機能により、利用アプリケーションに制約はない
  3. 初心者にも使いやすいメニュー画面や音声ヘルプを用意
  4. 視覚障害者に使いやすいコンソール環境を利用
  5. 初めて完全なマルチタスク・マルチユーザ機能が利用できる視覚障害者のためのシステム
  6. 日常必要なエディタやブラウザ、電子メールといったアプリケーションをはじめから備えたシステム

2.1 視覚障害者のための OS

視覚障害者のユーザに適した OS とはどのようなものであるか考察する。視覚障害者がコンピュータを利用するには以下のことが重要になる。

- 画面情報のわかりやすさ

  弱視者のユーザの場合、画面をカスタマイズしてコンピュータを利用している。しかし、弱視者のユーザの中にもスクリーンリーダーソフトを使用しているユーザは少

なくない。弱視者のユーザは画面情報をある程度認識できるため、スクリーンリーダーソフトのみに頼る必要はない。スクリーンリーダーソフトを画面情報を知るためのサポートとして使用すれば、晴眼者と同様にコンピュータを利用できると考えられる。

全盲のユーザ、あるい全盲にほぼ近いユーザが画面情報を知る方法として、スクリーンリーダーソフトの使用と点字ディスプレイの使用の 2 つが挙げられる。点字ディスプレイの使用は、点字が理解できるということが条件である。しかし、視覚障害者全体でも点字が使える人は少ない。そのため、点字ディスプレイを使用しているユーザは少ないと思われる。また、点字ディスプレイを使用しているユーザにおいて、点字ディスプレイのみを使っているのではなく、スクリーンリーダーソフトと組み合わせて使っているようである。これは、音声もあったほうが便利だからだと思われる。このことから、全盲者の場合、コンピュータ利用において音声の必要性は高いと考えられる。しかし、スクリーンリーダーソフトは画面情報のすべてを表せない。すべてというのは、対応していないアプリケーションソフトウェアでは読み上げはしてくれないということである。そのため、専用アプリケーションソフトを作る必要がある。また、CUI において可能であった文書作成やデータベース検索、通信、プログラミングはなんとか GUI において可能であるとしかいえない。

- コンピュータの操作性

  コンピュータの操作はキーボードまたはマウスで行なう。

  上でも述べたように、弱視者のユーザの場合、画面情報をある程度認識できる。マウスカーソルの位置も知ることができるから、アイコンへのアクセスも速くは無理だが可能である。

  しかし、全盲者のユーザが GUI の作業空間においてマウスカーソルの位置を把握するのは困難であると考えられる。しかしながら、キーボードのみで GUI 画面を操作することもまた困難である。例えば、アイコンへのアクセスはキーボードでは難しい。

以上のようなことから、視覚障害の比較的軽い弱視者のユーザの場合 GUI ベースの OS のほうが使いやすいと考えられるが、全盲者のユーザあるいは全盲にほぼ近い弱視者のユーザは GUI ベースの OS が適しているとはいえない。

そこで、そのようなユーザに対して CUI の OS である PC–UNIX ならどうであるかということを考察する。

GUI に比べると CUI の OS の方が、全盲者のユーザには画面情報はわかりやすいといえる。また、全盲者のユーザがコンピュータを利用するには、すべての動作をキーボードで行なう必要がある。

PC–UNIX は、 Linux 系と BSD 系に大別することができる。どちらの系統でも、いくつか商品化されているパッケージもあるが、フリーであるため、ユーザが自らの責任でではあるが、自由に入手、配付ができ、また、オープンソースで配付されるので、ユーザが自由にカスタマイズでき、GUI または CUI のどちらでも選択できる。

- Linux 系

  Linux には様々なディストリビューションがある。Linux では、デフォルトでアプリケーション、 X ウインドウシステム、ライブラリ、Linux カーネル、カーネルモジュールをまとめて使いやすくしたパッケージをディストリビューションと呼んでいる。Linux ディストリビューションは、パッケージの配付形式によって RPM 系、DEB 系、TGZ 系に分けられるが、 それぞれの違いは、アプリケーションのインストール方法、すなわちパッケージングシステムの違いに大別できる。様々なアプリケーションソフトがインストールされるため、最近はどのディストリビューションでも、最低 1G バイトのハードディスク容量が必要である。また、最近はどのディストリビューションも、インストール後に X ウインドウシステムが起動して MS–Windows と同様に GUI ベースの画面が表示されるため、視覚障害者には利用しにくい環境であるといえる。

- BSD 系

  BSD 系にも FreeBSD 、 NetBSD 、 OpenBSD などいくつかの種類がある。その中でも現在は FreeBSD の利用者が一番多いと思われる。Linux との違いは、インストール時にユーザが望むアプリケーションソフトだけをインストールすることができることである。もともと研究所や大学などに配付され開発された OS なのでコンピュータ初心者のユーザにとっては利用は困難であるとも考えられる。

2.2 マニュアルに関する問題

視覚障害者のコンピュータ利用の問題としてマニュアルに関する問題がある。マニュアルには、OS やアプリケーションソフトをインストールしたときに付属されるオンラインマニュアルと、市販されている書籍のようなペーパーマニュアルがある。

オンラインマニュアルは、音声化ソフトや、点字ディスプレイ、点字出力プリンタを使えば理解が可能である。

視覚障害者がコンピュータの取扱説明書や市販されているマニュアルを理解するには、ボランティアに点訳してもらったり、マニュアルを朗読してテープに録音してもらっている。しかし、この方法では、点訳や録音に時間がかかるため、すぐにマニュアルを入手できない。

他の方法として、MS–Windows では、フラットヘッドスキャナを利用して文字を読み取りコンピュータで文字認識させる OCR ソフトを利用する方法がある。以前は、専用の OCR 装置が多く、なおかつ高額なものだったが、 最近はスキャナに OCR ソフトが付属されているため、比較的安く利用できるようになったといえる。また、PC–UNIX では Linux に OCR プログラムとして xocr というフリーのパッケージがある。ただし、OCR を利用する上で重要なことは文字認識率の高さである。MS–Windows の OCR ソフトでの認識率は 99% 以上、 xocr での認識率は市販されている OCR ソフトよりも低い。

また、出版社の中には書籍のテキストデータをプレインテキストの形で提供している

ところもあるので、これを利用するのも 1 つの方法である。現在、ほとんどの出版社がコンピュータでマニュアルを作成しているが、出版社がどのような形式で作成しているかということが問題となる。例えば wv など MS–Word で作成した文書を PC–UNIX 上で html に変換するソフトウェアもあるが、文字化けすることもあり完全に変換できるわけではない。そのため、文書を特定のプラットホームやソフトウェアなどに依存しないように作成しなければならないし、それが実現できなければ CD–ROM や フロッピーディスクでの提供は難しい。

また、最近は Adobe 社 [10] で開発された PDF(Portable Document Format) が文書配付に使われることもあるが、これもテキストのみを抜きだして音声化することは難しい。

プレインテキストの提供は出版社の善意で行われているものであるため、利用者が少なかったり上記のような理由でプレインテキストの作成が難しい場合、プレインテキストの提供が無くなることもありうる。

以下は、ある出版社のプレインテキストの入手の条件と方法である。

1. 身体障害者手帳の写真が有る部分のコピー
2. 書籍を買ったという領収書
3. このデータは本人個人のみで使用する。
4. データは電子メールで受け取る (他の媒体での受け取りを希望の場合は要相談)。

しかし、上記のどの方法にしてもマニュアル内のグラフィカルな部分を視覚障害者に認識させることは困難であると考えられる。表やグラフに関しては、§2 でも述べたように、特殊なソフトウェアや点字ディスプレイを使えば知ることができるようであるが画像は音声や点字で表すことができないし、書籍は晴眼者を対象として書かれているため、テキストのみでなく画像を用いて解説している部分も多く視覚障害者がそれを理解するのは難しい。また、テキストのみであったとしても、テキストに書かれていることをそのまま音声にしても理解しやすいとは限らない。そのため、視覚障害者向けのマニュアルとして

1. 図や画像を用いない
2. 文章をなるべく簡潔にする
3. コマンド入力およびその出力結果の例を記述
4. 学習形式

を基礎にしたマニュアルについて考察する。PC–UNIX の利用において OS に関する知識やコマンドの理解が必要であるが、市販されている書籍では理解が難しく、またオンラインマニュアルの場合、各コマンドや関数ごとの情報といった必要最小限のことしか記述されていない。そこで、市販されているマニュアルとオンラインマニュアルの中間的なマニュアルが必要である。

マニュアルの構成としては PC–UNIX を使う上で必要となる基礎的な事項を記載するだけで、市販されている書籍のように詳しく記載する必要はないと考えられる。また、音声を利用する場合のために文章はなるべく簡潔にすることが必要である。

内容としては学習形式にする方法が考えられる。コマンドを覚えるには実際にコマンドを入力してそれに対してどのような出力結果が得られるか知ることで覚えていくことができると思われる。そのため、オンラインマニュアルのような各コマンドの説明も必要であるが、実際にコンピュータを利用する場合、他のコマンドも併用することがあるため、それら一連の動作を覚える必要がある。例えば、あるファイルに対して情報を見る、書き換えるという操作の他に必要でなくなったら消去する、あるいはゴミファイル専用のディレクトリを作りそこに移動させるといった操作を覚える必要がある。そこで、そのような関連した操作方法をマニュアル内で学習できるようにする必要があると考えられる。

また、学習できるようにするのなら、マニュアルはペーパー形式でなくオンラインマニュアルのようにディスプレイ上で見れるようにした方が良いと思われる。

3　視覚障害者の OS インストール環境

OS のインストールは、視覚障害者にとって容易なものではない。現在、インストールは、インストーラで半自動的に行なわれる。ほとんどのインストーラは対話型であり、出てくる質問事項に答えながらインストールの準備を進めていく。しかし、マウスを操作して設定していく GUI のインストーラは、視覚障害者には困難である。

そこで、CUI ベースのインストーラとして Linux 系、 BSD 系それぞれのインストール環境について考察する。

最近のインストールメディアとして、CD–ROM、フロッピーディスク、(モデムや LAN を経由できるなら) FTP や NFS のサーバホストを通じてのインストールが考えられる。これは、 Linux 系、 BSD 系のどちらでも同じである。

また、昨年末には、PC–UNIX (Debian/GNU) のインストール環境で点字ピンディスプレイを利用できるようになった。[13)]

- Linux 系のインストール環境

  ディストリビューションのインストーラは GUI と CUI のどちらでも選択できるようになっているが、最近では GUI メインのインストーラになりつつあるため、Linux のインストールにはマウスを利用できる。

  その中から RedHat Linux を取り上げる。RedHat Linux は手動インストールと KickStart と呼ばれる全自動インストールの 2 つのインストール方法がある。この機能を用いると、RedHat Linux のインストール作業の大部分 (あるいは全て) を自動化することができる。自動化できる処理には以下のものがある。

  – 使用する言語の選択
  – ネットワークの設定およびディストリビューションのインストール元の選択

- キーボードの選択
- (lilo 等の) ブートローダのインストール
- ディスクのパーティション設定とファイルシステムの作成
- マウスの選択
- X ウィンドウシステムのサーバの設定
- タイムゾーンの選択
- root のパスワード (の初期値) の選択
- インストールするパッケージの選択

KickStart システムは通常のインストール手順をスクリプト記述することによって可能にする。これは , 通常はキーボードから入力すべき情報を設定ファイルに書くことによって行うということである。しかし、 RedHat の起動フロッピーに KickStart の設定ファイルをコピーする作業が必要であり、その際、起動フロッピーの中の必要ないと思われるいくつかのファイル、または持っていないハードウェアのデバイスドライバを消して空きを作らなければならない。このとき、必要なファイルやデバイスドライバを誤って消してしまわないように注意する必要がある。

- BSD 系のインストール環境

  BSD 系の中から FreeBSD を取り上げて考察したい。FreeBSD のインストールは次のような手順で行なわれる。

  1. デバイス情報の編集
  2. 使用言語の選択
  3. キーマップの選択
  4. 各種インストールオプションの確認、設定
  5. インストール (初心者、クイック、カスタム) の選択

  5. で初心者を選んだときのインストール手順および設定 :

  1. FDISK でスライスの作成
  2. ブートセレクタの選択 (他の OS が存在する、あるいは 2 台目以降に FreeBSD スライスを確保した場合)
  3. パーティションの作成
  4. 配付ファイルの選択 (実際にインストールするものを選択する)
  5. インストールメディアの選択 (CD–ROM,FTP, フロッピー等)
  6. インストール実行

  また、インストール後には次のような環境設定が必要である。

- ネットワークインターフェイスの設定
- その他のシステムの設定 (タイムゾーン、マウス、管理者パスワードの設定、ユーザアカウントの作成など)

FreeBSD はテキストベースのインストーラである。上記のような設定はすべてテキストベースで行なわれる。そのため、 GUI のインストーラと比較して視覚障害者のユーザにはインストールしやすい環境であると思われる。

FreeBSD のようなテキストベースのインストーラでは curses ライブラリを用いているものが多い。curses ライブラリは、UNIX において、行ベースではなく、画面ベースのキャラクタ対話型アプリケーションを作成するためのライブラリである。UNIX の標準的な対話環境では、行単位の入出力しかサポートしていないが、エスケープシーケンスを駆使することでカーソルの移動や画面の任意の位置にテキストを表示できるようになる。そのため、1 つの画面の中にいくつもの設定に対する質問事項を表示することができる。また、 晴眼者がインストールする際に、入力設定を間違えた時にカーソルを自由に動かせることは便利である。

しかし、視覚障害者には curses ライブラリを用いたインストーラは向いていないと考えられる。また、視覚障害者のユーザのインストールに音声が必要であり、音声化ソフトを用いて文字列を順番に読むことで視覚障害者のユーザに画面情報を知らせるのだが、この場合、カーソルが自由に移動できるということはむしろ邪魔になると思われる。

音声化の容易さに関しては、文字列を 1 次元的に順番に流すだけならば音声化はしやすいと思われる。また、文字列をなるべく短くすれば、最初から読ませることしかできない場合でもそれほど面倒ではないと考えられる。また、設定に対する質問事項を画面に 1 つのみ表示するという方法を提示する。これは 1 つの設定ごとに画面を切り替えるという方法である。

以上はコンピュータを 2 台使うという従来の環境に対しての考察であるが、コンピュータ 1 台でのインストールを実現するにはインストーラを音声化する必要があり、そのためにはインストールディスクに音声化ソフトを組み込まなければならない。最近は CD–ROM 1 枚でインストーラの起動から配付ファイルのインストールまでできるようにもなっているが、従来インストールはフロッピーから行うもので起動フロッピーとディスク全体の内容を含んだフロッピーの 2 枚を使用していた。また、現在でも FTP や NFS のサーバホストからのインストール、あるいは CD–ROM ドライブからのブートができない場合、必ず起動フロッピーを作成しなければならない。インストールディスクに音声化ソフトを組み込むということは厳密に言えば起動フロッピーに音声化ソフトを組み込むことである。

コンピュータの起動は、システム ROM 内の小さなコードを実行し各種デバイスの初期化を行い、次にこのコードをディスク内にある OS の初期コードに制御を移すといった、より高レベルなソフトウェアを実行するために段階を追って初期化を行

う過程 (ブートストラップ) で行われるが、インストールでも同じようなことが行われる。

インストールの場合、起動フロッピーで立ち上げたミニカーネルをハードディスクにコピー、展開して OS をインストールするといった過程で行われる。このミニカーネル部分は小さいため、そこに音声化ソフトを組み込むことは無理である。

そこで、視覚障害者のユーザが一人でインストールできる方法の一つとして次のような方法を挙げる。

ユーザ一人一人に適したインストールディスクを提供する

これは、RedHat Linux の KickStart のように全自動でインストールを行なう方法である。インストールの際にユーザ自らが行なうさまざまな設定を、あらかじめ何らかの方法で入力してもらい、それらの設定をインストール中に選択しなくてもいいような自動インストールディスクを作り、提供するということである。入力の方法の 1 つとして FreeBSD の配付物を提供する FTP のサーバ上で各自のコンピュータに関する設定を入力してもらい、そこでインストールディスクを自動的に作成し提供するという方法が考えられる。この際問題となるのは、視覚障害者のユーザが、インターネットを使えて FTP サーバホストにアクセスが出来なければならないということである。

配付手段としてフロッピーディスクが考えられるが、容量の問題でインストールディスクが 2 枚以上になることが考えられる。全盲のユーザの場合、ドライブのアクセスランプを見ることができない。全盲のユーザがフロッピーディスクの入れ換えをする場合、フロッピーディスクドライブの回転の音が止まったのを確認してから入れ換えるといった方法で行なっている。そのため、インストールディスクの入れ換えをどのように知らせるかも問題になる。上でも述べたように、インストールディスクに音声ドライバを組み込むことは難しい。

そこで、他の配付手段としては CD–ROM が考えられる。最近は CD ドライブからブートできるようになっている。CD–ROM であれば容量に対する問題はなくなると思われる。そこで、インストールディスクとして CD–ROM ブート可能なものが作れればよいと思われる。

また、インストール中にエラーが発生した場合にどのように知らせるかということも問題になる。

3.1 インストールをするための質問事項

FTP サーバホストでの質問事項にどのようなものが必要か記述し、考察したい。質問事項の中には、Yes/No で答えられる選択肢、いくつかの中から選ばなければならない選択肢、あるいは自ら入力しなければならないものもある。

MS–Windows が導入されている。(y or n)

MS–DOS が導入されている。 (y or n)

上記の 2 つの質問事項は、下のパーティションの切り分けに関連するため必要であると考えられる。

パーティションはどのように切り分けますか。

この質問事項のところで、/ (ルート) 、スワップ、/usr、/var、/tmp のそれぞれの容量を入力するようにしたい。また、初心者のユーザのために、それぞれのパーティションの説明、例えば、

- スワップパーティション : 同時に実行しているプロセスが必要とするメモリの合計が実メモリのサイズを越えたときに、実メモリの内容を退避させるために使われる。
- /var パーティション : システムのログ情報や利用者宛のメール、プリンタに渡す印刷データのキューなど OS やその上で動くシステムアプリケーションが頻繁に読み書きするようなファイルを置くディレクトリ。OS の利用頻度や利用形態によっても変わる。個人的に利用するマシンの場合、5 ~ 10MB で十分。

のようなものも記述する必要がある。また、既存のインストーラには “自動レイアウトによるパーティション作成” という選択項目がある。バージョンがアップグレードするにつれて、自動レイアウトされるパーティションの容量は大きくなっている。

ハードディスクの容量の大きいものを使っているユーザは問題ないが、小さいものを使っていたり、他の OS を入れていて容量が少ない場合、自動レイアウトがうまく行かない場合があると考えられる。

ブートセレクタを作成しますか (y or n)

ハードディスク内に他の OS が存在する場合、または 2 台目以降に FreeBSD をインストールする場合に、起動時にどの OS を利用するか選択する必要がある。

ネットワークインターフェイスの設定を行ないますか (y or n)

OS の起動後でも行えるが、FTP や NFS のサーバホストからインストールする場合は必須である。設定を行う場合、ホスト名、ドメイン名などを入力する必要がある。

タイムゾーンの設定を行いますか (y or n)

PC(BIOS) が管理している時間が世界標準時か利用する地域のローカルタイムのどちらかを選択するのだが、今のところ日本の視覚障害者の利用に対してしか考えていないので、この設定はこちらで自動的に決めてしまってもよいかもしれない。

マウスを使用しますか (y or n)

ユーザアカウントを作成しますか (y or n)

管理者権限 (root) で作業することはファイルの破壊など様々な危険の原因になるので、ユーザアカウントをつくって、そのアカウントで作業したほうがよい。そのため、ユーザアカウントの作成の是非を質問するのではなく、下のように

> ユーザアカウントを作成してください
>
> 管理者権限 (root) で作業することは危険です

として、ユーザアカウントの作成を強制したほうがよいかもしれない。

> [root] のパスワードを設定してください

パスワードは 5 文字以上で、大文字と小文字を混在させるか、記号や数字を混ぜる必要があります

4 視覚障害者の PC–UNIX 使用環境

視覚障害者が PC–UNIX を使用することの利点、問題点などについて考察する。

4.1 PC–UNIX における日本語環境

UNIX は、もともと英語用に設計されている OS である。そのため、FreeBSD や Linux で日本語を取り扱うには、様々な設定を行なう必要がある。PC–UNIX において日本語環境を取り扱う環境は以下の要素で成り立っている。

- 画面に日本語を表示できる
- 日本語を入力できる
- 日本語文書を印刷できる
- 日本語で作業を行なえる

PC–UNIX における日本語環境の現状は Fig.1 のようになる。
以下で詳しく述べる。

- PC–UNIX 上の日本語表示

  一般に、PC–UNIX の環境でコンソール上に日本語を表示することはできない。しかし、[kon] というプログラムを使用することで、簡単に漢字を表示することができる。

  X ウインドウシステム上では、[xterm] というターミナルエミュレータを標準で使用するが、日本語を表示することができない。

  そこで、 X ウインドウシステム上で日本語を表示させるには、[kterm] というターミナルエミュレータを使用する。

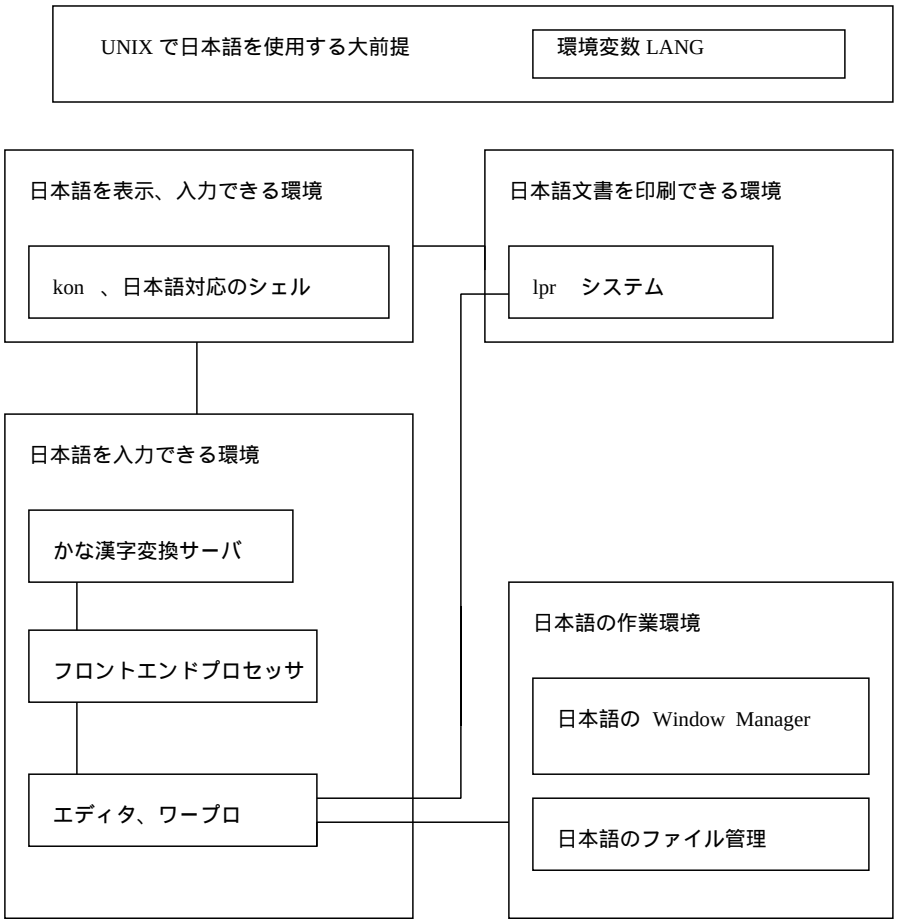

Fig. 1 日本語環境の現状

- PC–UNIX 上の日本語入力

  kon や kterm 上で日本語を表示させるのは可能になるが、そのまま日本語を入力しようとしても、シェルが日本語を通さないため文字化けを起こしてしまう。そのような場合、日本語対応シェルを使用する必要がある。日本語対応のシェルとして、tcsh 、 bash がある。

- かな漢字変換エンジン

  日本語を入力するには、かな漢字変換エンジンが必要である。かな漢字変換エンジンは、入力されたローマ字を漢字に変換するためのプログラムである。

  以下に主な、かな漢字変換エンジンを記す。

  - Wnn

    Wnn には Wnn4.2 と Wnn6 があり、かな漢字変換サーバ部分の jserver にフロントエンドプロセッサ (FEP) 、辞書を含めたものの総称である。

Wnn4.2 は FreeBSD のパッケージにも含まれているフリーのかな漢字変換エンジンである。

Wnn6 はフリーウェアではなく、オムロンソフトウェア株式会社の商用のかな漢字変換エンジンである。Wnn4.2 よりも辞書が多く、高い変換効率を誇る。しかし、メモリの負担が大きいため非力なマシンでは動作が遅くなる。

- Canna

  Canna は NEC が作ったフリーのかな漢字変換エンジンである。Wnn4.2 や Wnn6 よりも動作が軽く、少ないメモリで動くという利点がある。変換効率は Wnn4.2 とあまり変わらないようである。

- SKK

  SKK は、Mule 上でのみ使用できるフリーのかな漢字変換エンジンであり、以下のような他と違ういくつかの特徴を持っている。

  * 口語体や方言を変換できる
  * 変換に失敗すると自動的に辞書登録モードに入る
  * [File] を [ファイル] のように、英語の綴りから変換できる

- FEP (フロントエンドプロセッサ)

  FEP は、ユーザとアプリケーションの間で、入力の制御を行なうプログラムである。ユーザは、 FEP の操作に合わせてローマ字入力などを行なうことで日本語を扱うことができる。

  また、 X ウインドウシステム上で使用できる FEP を XIM (X Input Method) という。XIM とは、元は X ウインドウシステム上で行なう文字入力の決まりのようなものであった。

  FEP には、 Wnn で使用される uum や Canna で使用する canuum 、 Wnn 専用の XIM である Xwnmo 、そして Kinput2 がある。kinput2 は Wnn 、 Canna の両方に対応した XIM である。

- エディタ

  エディタには、大きく分けてモードありエディタとモードレスエディタがある。モードありエディタとは、複数の機能を持ち、文字の入力を行なう場合と入力されたテキストの編集を行なう場合などとで、エディタの機能モードの切り替えが必要なエディタである。その結果、同じキーを入力しても、モードにより効果か異なることになる。一方、モードレスエディタでは、通常のアルファベットキーを押せば常にテキスト文字の入力が行なわれる。PC–UNIX で使用するエディタでは、 [vi] と [Mule] の 2 つが代表的である。

  - vi

    [vi] は、非常に動作の軽いエディタである。vi エディタは、モードありエディタであり、次のような 3 つのモードを持つ。

* テキスト入力モード
* コマンド (編集) モード
* ex モード

テキスト入力モードは、文字入力を行なうモードであり、文字入力以外にはカーソル移動機能がある程度である。コマンドモードでは、どのキーを押してもテキスト文字の入力は行なわれず、文字、語や行の削除や文字列の探索といった編集を行なうモードである。ex モードは、ラインエディタモードである。

3 つのモードの関係を以下に示す。

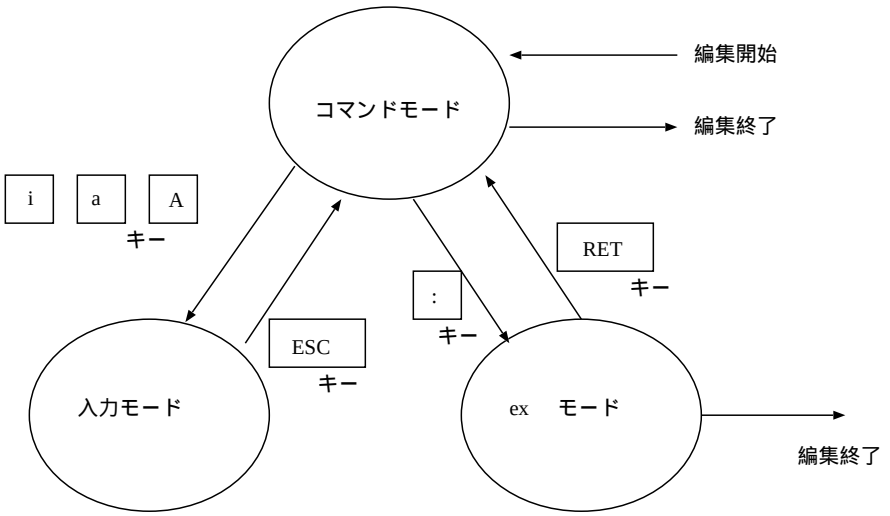

Fig. 2 vi のモード間遷移

– Mule (Emacs)

[Mule (MUltiLingual Emacs)] は、 GNU Emacs という非常に高機能なエディタを多国語対応にしたもので、多言語 Emacs とも呼ばれている。

モードレスのフルスクリーンエディタであるが、カーソルの移動や文字の挿入、削除といったエディタ操作のすべてが、 Emacs Lisp と呼ばれる Lisp 言語のコマンドとして実装されており、それらをリアルタイムに Lisp インタープリタ上で実行するというアーキテクチャが採用されている。これにより、 Emacs Lisp でプログラムを記述することで Emacs の動作をどのようにも変えることができ、Emacs をベースにしたメールリーダやニュースリーダなどが Emacs Lisp プログラムで作成されている。また、マルチスクリーン機能があり、複数のファイルの編集なども容易である。

Emacs の機能をまとめると次のようなものが挙げられる。

* マクロ機能 (拡張機能)
* テキストの入力、編集

- 複数の書式モード
- マルチウインドウ
- マルチ編集バッファ
- 編集バッファの印刷
- ディレクトリ編集
- UNIX コマンドの実行
- 編集バッファのコンパイル
- メールの処理
- ヘルプ機能
- オンラインマニュアル機能

このように、エディタとしてだけでなく、ほとんどすべての UNIX 操作が Emacs 内で実行可能である。

4.2 視覚障害者のための日本語環境

視覚障害者の日本語環境として、 Emacs を挙げる。視覚障害者がコンピュータを利用する目的として、電子メールやインターネットが挙げられる。上記のように、Emacs はエディタとしてだけでなく、メールの閲覧、インターネットも可能である。

視覚障害者が文字を入力する際の問題点として漢字変換がある。例えば、「感じ」と入力しようとした時、その変換候補に「漢字」「感じ」「幹事」のようにいくつかの候補が出てくるのだが、「感覚の感」や「敏感の感」といった音声での確認を変換する時点では行うことができない。そのため、視覚障害者のユーザは文章を書いてから音声で読み上げ校正を行っている。文章の漢字変換は、かな漢字変換エンジンの性能次第だが、ユーザは望む文章への変換を自分で手直しすることが多々ある。そこで、漢字変換の時点で音声読み上げによって望む語句への変換を行なうには、かな漢字変換エンジンを音声化する必要がある。

視覚障害者が Emacs を使用する時の問題点として、ここでも音声化の問題がある。Emacs には、 Emacspeak という Emacs のすべてを音声化するシステムがあるが現在は英語版しかない。しかし、日本語が使えるように開発しているプロジェクトもある。[11] このプロジェクトでは、英語と日本語を平等に扱えるように拡張すると共に、Linux 用の使いやすい音声エンジンの開発に取り組んでいるが FreeBSD 用の音声エンジンの開発は考えていないようである。

そのため、現状において PC–UNIX 上で Emacs を使うには、音声化ソフトの利用という方法を考えるしかない。コンピュータを 2 台使用して MS–DOS の音声化ソフトを使用する方法もあるが、それではコストがかかるのでコンピュータ 1 台で使用できるようにしたい。

Emacs での音声化の方法は §4.3.1 で述べるシェルの音声化に頼るものである。Emacs に限らず、アプリケーションソフトはプログラムであり、コマンドを入力することによって、アプリケーションソフトは起動する。

シェルを親プロセスとすると、アプリケーションソフトは親プロセスから派生した子プロセスといえる。親プロセスであるシェルを音声化した場合、子プロセスであるアプリケーションソフトも音声化されると考えられる。

4.2.1 シェル

ユーザが打ち込む命令 (コマンド) を読み込み、解釈して、それに応じた動作を実行させてくれるプログラムのことをコマンドインタプリタと呼ぶ。UNIX では伝統的にコマンドインタプリタのことをシェル (shell) と呼んでいる。シェルは、ユーザが対話形式でプログラムの実行を制御できるプログラムであり、ユーザがログインするとシステムによって最初に起動されるプログラムがシェルである。そのため、シェルは、OS の機能を利用するためのユーザインターフェイスプログラムといえる。他の OS では、コマンドインタプリタはシステムに組み込まれた特殊なプログラムだが、UNIX ではシェルはプログラムと同じように書き換えることができる。

シェルには対話型シェルと対話型シェルでないシェルがある。対話型シェルは、ユーザにコマンドライン=インターフェイスを提供し、ユーザが入力した文字列を解釈してコマンドを実行する。一方、対話型シェルではないシェルは、ファイルや引数を解釈してコマンドを実行する。

シェルには、大きく分けて Bourne シェル系と C シェル系の 2 つの系列がある。以下に FreeBSD で利用できる主なシェルを述べる。

- Bourne シェル系
  - sh : Bourne シェル
    sh は UNIX の標準のコマンドインタプリタで、原作者の名前から Bourne シェル呼ばれる。対話型シェルとしての使い勝手が悪いため、シングルユーザモード以外では対話型シェルとして使われることはほとんどない。プログラミング言語としての機能は強力で、シェルスクリプトで主に利用されている。
    FreeBSD のシェルは、厳密には Bourne シェルではなく ash と呼ばれるシェルである。
    そのため、コマンド行編集、コマンドヒストリ、ジョブコントロールなどの Bourne シェルにない機能を利用できる。
  - ksh : Korn シェル
    David Korn の作った Bourne シェル系のシェル。Bourne シェルにヒストリ機能などの改良を加えたもの。AT & T 社の有償ソフトだったため、あまり普及はしなかった。フリーソフトウェア版の pdksh というものもあるが、 ksh の機能をフルにサポートしているわけではない。
  - bash : Bourne–Again シェル
    bash は GNU プロジェクトのシェルで Bourne–Again SHell の略である。ksh(Korn シェル) や csh の機能を取り入れてある。

Linux ディストリビューションの多くは、bash が標準装備されている。

- zsh

  他のシェルのような様々な機能を取り込んでいるのに加え、独自の機能も充実している。Bourne シェル系に属しているが、csh 系からの乗り換えが容易なように配慮されている。

  zsh の機能の中で特筆すべきなのは、カスタマイズ可能な強力な補完機能とコマンド行エディタである。

- C シェル系

  - csh : C シェル

    カリフォルニア大学バークレー校 (UCB) で開発された C シェルは、Bourne シェルとはかなり設計を異にしており、コマンドヒストリ、エイリアス、ジョブコントロールなどの機能を備え、特に対話的機能に優れており、対話型シェルとして広く使われるようになった。

    プログラム言語としては、C に似た文法を採用しているのが特徴で、Bourne シェルとの互換性はない。

  - tcsh

    csh を大幅に機能アップしたシェルで、csh との互換性を保ちつつ、コマンド行編集、入力の補完をはじめとする多彩な機能を備えている。プログラミングの機能も豊富だが、特に対話型シェルとして非常に使いやすいものになっている。

    最新の FreeBSD のバージョンでは tcsh を csh としている。

4.3 シェルの機能

bash や tcsh などで使える機能は次のようなものである。

- コマンドライン編集

  Emacs と同様な操作で文字編集やカーソル移動、インクリメンタル検索ができる。キーバインドを vi 風にカスタマイズすることも可能。

- ヒストリー (コマンド履歴)

  それまでに実行したコマンド文字列を参照できる。また、その文字列の一部を編集する機能もある。

- 自動補完

  コマンド名、ファイル名、ホスト名、ユーザ名、シェル変数や環境変数名をキー 1 つで補完する機能。

Table 1 シェル機能の比較

	sh	csh	tcsh	ksh	bash	zsh
ファイル名の展開	○	○	○	○	○	○
ファイル名の補完	×	○	○	○	○	○
コマンド行の履歴	×	○	○	○	○	○
コマンド行の編集	×	×	○	○	○	○
シェルスクリプト	○	○	○	○	○	○
コマンドの別名	×	○	○	○	○	○
リダイレクション	○	○	○	○	○	○
関数	○	×	×	○	○	○
ジョブ制御	×	○	○	○	○	○

- ジョブ制御 (job control)

  プロセスの実行を一時的に停止させたり、プロセスをバックグラウンド (裏) で実行したり、バックグラウンド=プロセスをフォアグラウンド (表) にもってきたりすることで、1 つの端末 (ウインドウ) で多くのプロセスを制御できる。

- エイリアス (別名定義)

  コマンドが長かったり、つねに指定する引数がある場合など、たくさんの文字を入力するのは面倒であり、間違えやすくなる。エイリアス (alias) とは、これらを別名定義しておき文字入力を減らすことができる機能。

- ファイル名の展開

  ユーザが入力したコマンドに ~ や * 、? などの特殊文字を見つけると、あるルールに従ってファイル名に展開する。これによって、ユーザはファイル名やパスを簡単に指定できる。

- リダイレクション

  コマンドへの入力、コマンドからの出力を任意に変更する機能。

4.3.1 視覚障害者の使いやすいシェル

視覚障害者に使いやすいシェルを考察する。

Table.1 から bash や zsh はシェル機能が豊富であり、また、tcsh にしても関数機能が使えないだけである。このことから、シェルの機能による差異はそれほどないことが分かる。

しかし、現在多くのユーザに利用されている bash、tcsh といったシェルは、晴眼者のユーザにとって使いやすいように改良されてきたシェルであるため、視覚障害者のユーザが使いやすいシェルとは言えない。

例えば、作業画面に表示される # 、MS–DOS なら A> などのプロンプトは、本来、晴眼者に入力可能かどうか示すものである。もし、全盲者のユーザが入力可能かどうか知るためには、プロンプトの代わりに何らかの音を出力させることが必要になる。

音声を出力させる方法として、シェル自体を音声化する方法がある。PC–UNIX においてシェルはプログラムのように書き換えることができるので音声デバイスを組み込むようにプログラムを書き換えればよい。しかし、シェルを音声化しただけでは使いやすくなるとは言えない。

そのため、現在の環境の音声化とは別の方向性を考察する必要がある。

しかし、視覚障害者にシェルが使いやすくない理由は、対話型であることやシェルの機能に問題があるというわけではないと考えられる。むしろ、補完機能やエイリアスのようなシェルの機能があったほうが便利である。

また、現在のような対話型シェルが利用される前はユーザがプログラムのロード、入出力の割当て、プログラムの実行という作業を逐次的に実行する “バッチ処理” が基本であった。これは現在のシェルスクリプトを書くということと同じである。そのため、入力に対して何らかの反応がある対話型シェルのほうが視覚障害者のユーザには使いやすいと考えられる。

つまり現在のシェルの問題点は、音声化をするとファイル名などの出力が理解しにくいということである。例えば、現在ある音声読み上げソフトに英数字を含んだファイル名を読ませたなら英数字を 1 文字 1 文字読んでいくと思われ、これでは何というファイル名か理解するのに時間がかかる。そのため、シェルを音声化するにはファイル名を日本語で付けることが必要である。さらに、別の方法としてシェルを日本語化することも考えられる。

また、視覚障害者にとって対話型シェルの使いやすさは入出力のデバイス次第であるとも考えられる。シェルは入力デバイスとしてキーボード、出力デバイスとしてコンソール画面をデフォルトで使用する。次世代インターフェイスとして AUI(Auditory User Interface) が開発されているが、その入出力デバイスはヘッドホンマイクを使用する。これによりディスプレイの必要性はなくなると考えられるがキーボードの必要性はなくならないと思われる。

4.4 視覚障害者に使いやすい PC–UNIX 環境

ここまで、PC–UNIX を使うための基となる日本語環境とシェルについて述べてきた。ここからは、視覚障害者が実際に PC–UNIX を使うとき、どのような環境が良いか考察する。

- ウインドウ間移動

晴眼者の場合 X ウインドウシステムを利用した作業が容易であるが、視覚障害者にはウインドウ間移動が困難なため、視覚障害者のユーザが作業を行なうには

- 入力操作がキーボードのみで可能
- コピー & ペーストが簡単に行なえる

音声を出力させる方法として、シェル自体を音声化する方法がある。が重要な要素であると考えられる。

PC–UNIX においての作業はコンソール画面上ですべて行なうことができる。日常、コンピュータで行なう作業といえば、文書作成、電子メールやインターネットというようなものが挙がるが、これを 1 つのコンソール画面上で行なうには画面を切り替える必要があり、それを実現する方法として screen というフリーのソフトウェアを使う方法がある。sceen は、複数のプロセス間 (特に対話型シェル) で物理的なコンソールを複数個使えるようにする、全画面ウインドウの管理ソフトウェアである。各仮想コンソールは、エスケープシーケンスに加えて、行の挿入、削除や複数の文字集合のサポートといった制御機能をいくつか持っている。また、それぞれの仮想コンソールで履歴バッファをスクロールする機能や、異なるウインドウ間でテキストをやりとりできるコピー & ペースト機能がある。

screen は呼び出されるとウィンドウを 1 つ生成して、その中でシェル (または指定されたコマンド) を実行する。そしてユーザが通常通りプログラムを実行できるようにするため、ユーザから見えなくなる。その後は、新しい (全画面) ウィンドウを生成してそこで他のプログラム (別のシェルも含む) を実行すること、現在あるウィンドウを削除すること、ウィンドウのリストを表示すること、ログ出力を有効または無効にすること、ウィンドウ間でテキストのコピー & ペーストを行うこと、履歴をスクロールバックして見ること、自由にウィンドウを切替えること等がいつでもできる。プログラムが終了すると、 screen はそのプログラムが動作していたウィンドウを削除する。このウィンドウがフォアグラウンドにある場合は、表示は以前のウィンドウに切り替わる。残っているウィンドウが無くなると screen 自身も終了する。

全盲の視覚障害者のユーザの中には screen を使用しているユーザが多いことから、screen は有用であると考えられる。

5 おわりに

今回の研究では、視覚障害者の PC–UNIX を導入する際の問題点ならびに PC–UNIX を利用する利点について考察した。その結果、全盲者や全盲に近い弱視者ならば、PC–UNIX を利用したほうが便利であると考えられる。残念なことに今回は PC–UNIX 導入に関する考察のみで終ってしまったが、実際に OS の自動インストールディスクの作成、現在の環境の音声化を行なえばまだまだ多くの問題が見えてくるかもしれない。また、PC–UNIX におけるシェルや日本語環境の現状と問題点、その解決方法として視覚障害者にとって使

いやすい環境について考察したが、現時点の問題点は使いやすい音声化ソフトがないこと、音声化してもシェルは使いやすくならない可能性があるということがわかった。そのため、使いやすい音声化ソフトの開発およびシェルの音声化にともなう使いやすい対話型環境の構築が必要であり、今回は対話型環境を使いやすくする一例としてシェルの日本語化を挙げたが別の方法を考察することも今後の課題である。

参考文献

[1] Welcome to Katsuaki Watanabe's world! : http://www.watakatsu.com/

[2] 画面を見ないで Windows パソコン : http://www.lares.dti.ne.jp/ tomof/

[3] あさだたくや 天川修平 衛藤敏寿 浜田直樹 細川達己 三田吉郎
: FreeBSD 徹底入門 (翔泳社 1997)

[4] RedHat Linux KickStart HOWTO
: http://www.linux.or.jp/JF/JFdocs/KickStart-HOWTO.html

[5] アスキーデジタル用語辞典 : http://yougo.ascii24.com/

[6] 荒井誠 佐々木勝彦 山口聡 : 日本語環境構築入門 (秀和システム 1998)

[7] JARVI-ML Technical Digest : http://www.twcu.ac.jp/~k-oda/VIRN/JARVI-MLTD/index.html

[8] パソコンボランティアスキルアップセミナー : http://www.kawa-v.psv.org/skillup/

[9] Blind + Linux =LINUX : http://leb.net/blinux/

[10] Adobe Systems Incorporated. : http://www.adobe.co.jp/main.html

[11] Bilingual Emacspeak Project : http://www.argv.org/bep/

[12] ゼータビッツ株式会社 : http://www.ZetaBITS.com/index.html

[13] BRLTTY 対応 potato インストールイメージのインストール
: http://www.topstudio.co.jp/ kmuto/debian/column/brltty.html